Proceedings of Machine Learning Research 1–11

DLMI Kaggle Challenge
Prostate Cancer Diagnostic using Histopathology Images

Kaggle Team: SM-EG

Siwar Mhadhbi SIWAR.MHADHBI@TELECOM-PARIS.FR
Eya Ghamgui EYA.GHAMGUI@TELECOM-PARIS.FR

Introduction

Prostate cancer is a serious disease worldwide. In fact, men of all ages can be affected by this deadly disease. Recently, the use of new technologies (e.g., computer vision techniques) has been expanded to help doctors make the appropriate decisions. Early identification plays a vital role in the diagnosis and prognosis of prostate cancer. The International Society of Urological Pathology (ISUP) modified Gleason grades is one of the most powerful prognostic indicators for clinically localized prostate cancer, and one of the most critical factors in determining the management of these patients. However, it is quite difficult for pathologists to provide indicator grades quickly and efficiently. Manual processing is time consuming and delays treatment. In addition, it is not cost-effective. Thus, deep learning techniques can provide a better ISUP grade while reducing human error by increasing accuracy at all locations.

In this challenge, we will build deep learning models to classify prostate histopathology images. Since prostate histopathology images are highly imbalanced between prostate tissue and background, a patch selection method is needed to reduce the computational time and the impact of background on model performance in order to extract patches only on the informative and most representative regions of where cancer can be detected. In addition, the intensities of the histopathological images vary from one image to another. Thus, a normalization step is mandatory. Then, we proposed 3 different classification models. The first one is a modified ResNet18 pre-trained model. The second one is a concatenated tile pooling architecture using the Resnet18. The last model is an attention-based deep Multiple Instance Learning (MIL) architecture using ResNet18. In each model, we varied the learning method and the way of handling patches. In a second step, we investigated the effect of data augmentation on the performance of the models. In addition, we explored Curriculum Labeling as a semi-supervised method. We added an ablation study to tiles extraction, data normalization, and the loss function in which we studied the role of ordinal regression loss. Finally, we found that the best score is obtained for the modified ResNet18 with an AUC score on the Leaderboard equal to 0.9117.

1. Architecture and methodological components

In this section, we will discuss the methodology we followed to address the underlying histopathology problem. We will cover two main aspects of our methodology. The first one concerns how we approach the data set, and the second one concerns the modeling part, particularly which architectures we propose.

1.1. Methodological data pre-processing

The data set is a proportion of microscopic scans of prostate biopsy samples. We are provided with 340 whole slide pathological images in TIFF format with three levels of high resolution where 338 images are provided with their corresponding masks. Figures 1 and 2 represent a few examples of, respectively, the WSIs and the masks. We notice that the WSIs have large areas of empty space that do not contain the prostate tissue. This can be an issue as conventional approaches are not very efficient for this particular data. Actually, the size of provided images vary in a wide range. As a result, the input images are deformed in a not consistent manner upon rescaling that limits the ability of the model to learn. Furthermore, the large empty areas only result in inefficient use of memory and GPU time.

In literature, researchers have been working on the automated processing of these types of images such as the identification of biological regions, for instance tumor, stroma, and necrotic tissue. In our work, we proceed by eliminating these blank regions and keeping only prostate regions for each WSI. In our research, we got inspired by the kernel (1) and we proceeded by tiling the images. We will discuss in detail in the ablation study in section 2.1 how we applied the masks for the extraction step, as well as the appropriate number and size of tiles we chose.

Another interesting aspect we noticed is the difference in color of images (cf. Figure 3). Standard normalization does not work in this case which would bias the model. Thus, we thought of applying "stain normalization" inspired by (Vahadane et al., 2016). For practical purposes, we saved the extracted and preprocessed tiles once for all in (2) for easier access. The images were saved in TIF format as it is lossless, which means that we do not lose any quality after saving and editing the images. Also, TIF images are larger and have a higher resolution than JPEG.

1.2. Methodological modeling

- 1^{st} **Architecture: ResNet18**

  First, we used a modified ResNet18 architecture. This model consists of ResNet18 layers pre-trained on the ImageNet dataset, but we replaced the last layer with two fully connected layers. The first one has a hidden size of 64 and the second is the classification layer. Next, we trained the model using cross-entropy loss and the SGD optimizer with a learning rate equal to 10^{-2} and momentum equal to 0.8 for 10 epochs and a batch size equal to 17. The input to this architecture is a concatenation of 20 tiles extracted from the prostate images as described in section 2.1. These tiles are concatenated into five rows and four columns to form an image of size 640×512. On the other hand, we studied the performance of this model with and without data augmentation. To do so, we applied two data transformations which are the horizontal and vertical random flip.

- 2^{nd} **Architecture: Pooling ResNet18**

  The second architecture is based on the concatenated tile pooling approach. This model takes as input the same images as the previous architecture. These images are then passed through feature extraction layers built from a pre-trained ResNet18. It is applied on each tile of the image seperately. Then, the outputs of this convolutional part are concatenated into a single large map for each image preceding the adaptive concatenated pooling layer and the fully

connected head. Since all spatial information is removed by the pooling layer, the Concat Tile pooling approach is nearly identical to passing an entire image through the convolutional part, excluding predictions for near-empty regions, which do not contribute to the final prediction. Figure 4 illustrates the architecture chosen for the second model. We used, for learning, the cross-entropy loss and the SGD optimizer with a learning rate equal to 10^{-3}. The learning is performed in two stages. In the first 10 epochs, the model is trained with frozen ResNet18 pre-trained layers. Then, in the next 10 epochs, we unfreeze the entire architecture and the model is further tuned. We also studied the performance of the model in the cases with and without data augmentation.

To go further, we wanted to perform a semi-supervised approach. To do so, we loaded from the Panda challenge (3) 300 histopathological images (this number is related to the limitation of the resources). Then, we applied the same data preprocessing steps and the same tile extraction methods. After that, we implemented the Curriculum Labeling approach (cf. Figure 6). In this part, we were inspired by the idea of (Cascante-Bonilla et al., 2020). Their method is based on a pseudo-labeling approach that consists in using a model trained on labeled data to generate pseudo-labels for the unlabeled data set. Then, the most reliable pseudo-labels (with respect to a chosen threshold) are added to the labeled data set. After that, the process is repeated by training a new model from scratch on the combined labeled data set and generating pseudo-labels for the remaining unlabeled samples. The process stops when the number of total labeled samples exceeds a chosen number, which is in this case 450. To do this, we used the previous concatenated tile pooling architecture and the same strategy to train the model with a batch size equal to 17, a number of epochs equal to 8×2 and the SGD optimizer with a learning rate equal to 10^{-3}. We set the pseudo-labels selection threshold to 0.75.

- 3^{rd} **Architecture: MIL ResNet18**

  Our third architecture relies on a recently well known method in histopathology problems, which is Multiple Instance Learning (MIL). MIL is a variation of supervised learning that is more suitable for pathology applications. The standard MIL assumption consists in assuming that diseased tissue samples have both abnormal and healthy regions, while healthy tissue samples only have healthy regions. To adapt the approach to our problem, we divide WSIs into tiles where each collection of tiles, denoted as a bag, is labeled with its corresponding ISUP grade. In the training phase, the model should be learning which tiles are the cause of the severity of the disease. We thus implemented an attention-based deep MIL model for prostate cancer diagnosis, which is a modified version of the model used by (Ilse et al., 2018). The overall architecture is represented in Figure 5. It is composed of two feature extractor blocks: the first block is composed of a 3×3 convolutional layer followed by a ReLU activation and a MaxPooling layer with stride 2. The second block is composed of the $2 - 6$ layers of ResNet18. Next, a self attention block is applied. It is composed of a first fully connected layer with 64 units, a Tanh activation and a second fully connected layer also with 64 units. Finally, a classifier block, composed of a fully connected layer with 6 units, is added for the classification task. Unlike previous approaches, we feed the MIL model with one collection of tiles at each time (i.e., one prostate image), which is represented by a batch size equal to 1. We use the same loss function as the previous models and the same learning process. As for Pooling ResNet18, we also tested the MIL approach with both data augmentation and semi-supervised learning.

2. Model tuning and comparison

2.1. Ablation study

For the tile extraction process, we set the tile size to 128×128, as it seems to match the tumor regions well in this case. As for the number of tiles, due to the limitation of memory capacity, we cannot consider all tiles. Therefore, we have to proceed to the selection step. First, in order to take into consideration most of the tiles, we proceeded by selecting all 90 pre-processed tiles and then randomly choosing 20 tiles to feed to our models. However, this method led to poor results. This can be explained by the fact that the chosen tiles are not accurate. So, we switched to extracting the first 20 tiles from the region of interest. Prior to the extraction step, we applied the provided masks on the 338 images to further highlight the boundaries between the prostate region and the background and to remove any uninteresting regions from the background, such as spots. However, in doing so, we noticed that the stains problem remains. Initially, we thought that after applying the masks and extracting the tiles, we would select the tiles with low tissue content because the empty space is represented by white pixels. Nevertheless, we noticed that for some images, the tiles are extracted from the stained regions. When examining the masks, we noticed the presence of many false positives, the stained regions are falsely labeled as prostate regions, as shown in Figure 7. Hence, we thought that instead of selecting the low-tissue regions, we apply masks by setting the background pixels to black instead of white. After that, we convert the images from RGB to HSV color space and then we select the high-tissue tiles in the Hue channel. A comparison between the results of these two approaches is shown in Figure 8. For images without masks, we considered replacing the white pixels with black pixels. However, this would not only change the background color, but also the pixels inside the prostate regions, which we do not wish. Thus, we kept the first approach for WSI without corresponding masks.

When choosing the loss function, we thought that the conventional losses used for classification problems do not match our intention in the prostate cancer diagnosis problem. Indeed, conventional losses, for example cross-entropy, do not take into account the fact that the penalty incurred for errors must increase with the order of the labels, i.e. with the degree of severity of the prostate cancer. It is clear that our problem is a discrete ordinal regression, so, to solve this degree loss problem, we thought of using a loss function for preference levels as defined by (Rennie and Srebro, 2005). However, in our case, the use of ordinary loss did not work well. The algorithm appeared to be biased toward two distant degrees that are very high severity with grade 5 and very low severity with grade 1, both on the validation set and the test set, resulting in very poor performance, although it worked perfectly on the training data by achieving a perfect AUC of 1 and 100% accuracy. This aspect of biased predictions initially seemed to be related to the overfitting problem, but even when varying the model hyperparameters, architecture blocks, or even the feature extractor backbone, the model performance remained very poor. Therefore, we finally chose the conventional cross-entropy as the loss function for the three models.

2.2. Experimental Results

For all models, we first trained the architecture on 80% of the data and evaluated the remaining data in order to fine-tune the hyperparameters. Then, to create the submission file, we trained the model on the entire data set. For evaluation, we calculated the AUC and Balanced Accuracy metrics

and plotted the corresponding confusion matrix. We also plotted the evolution of the AUC and the Balanced Accuracy of the training and validation in function of the number of epochs.

- 1^{st} **Model: ResNet18**

  For the modified ResNet18, we obtained an AUC score of 1 on the training data, an AUC score of 0.7599 on the validation data, and an AUC score of on 0.9117 on the test data. Furthermore, we can notice that the AUC and Balanced Accuracy curves (cf. Figure 9) are overfitting even though we tried to adjust the model hyperparameters (learning rate, optimizer type, and momentum value). After adding the data augmentation, we can remark that the model does not show overfitting (cf. Figure 10). In fact, the gap between the training and validation curves is reduced for both metrics. The training AUC reaches 0.9777 and the validation AUC 0.9225. However, this method reduced the performance of the model on the leaderboard by 0.1976.

- 2^{nd} **Model: Pooling ResNet18**

  For the concatenated tile pooling model, the training settings are chosen to have the best performance on the validation dataset chosen to be 20% of the data size. In addition, the SGD optimizer was found to have better performance compared to Adam and AdamW for a best learning rate equal to 10^{-3}. As a result, we obtained the same training AUC, but a lower validation AUC (0.7025) compared to the previous model. On the leaderboard, this model provides a weaker performance (0.8089) than the previous model with a decrease in performance of about 0.1687. Moreover, the training and validation curves show a remarkable overfitting (cf. Figure 11). In this case, data augmentation improved the learning of the model. There is no remarkable gap between the training and validation AUC and the Balanced Accuracy curves (cf. Figure 12). Add to that, its validation AUC is promising (0.8918) and it gives a score of 0.8243 on the test data set.

  For the semi-supervised model, we successfully added the 300 pseudo-labeled data to the training data. Training the model on the new labeled data set gave us a training AUC of about 0.8669. But, only 0.659 on the validation data and 0.70897 in the leaderboard. Thus, we can say that the model does not generate consistent labels. Indeed, these labels do not add any additional information about the structure of the data and do not help the model capture deeper details about the disease. We can say that, in this case, this semi-supervised approach regresses the performance of the model and distorts its predictions.

- 3^{rd} **Model: MIL ResNet18**

  For the ResNet18 MIL model, by training the model for 30 epochs with a batch size of 1, SGD optimizer with a momentum of 0.8 and a learning rate of 10^{-3} which is reduced by 0.1 for every 10 epochs, we obtained perfect performance on the training set with an AUC of 1 and 99.44% accuracy versus poor performance on the validation set with an AUC of 0.56 and 24.73% accuracy, which is slightly better than a random classifier model. Figure 13 shows the performance of the model on the training and validation sets. Since we observe a remarkable overfitting, we then proceed to augment the data to vary the training samples with random horizontal and vertical flips. In this way, we obtained exactly the same performance on the training set with an improvement on the validation set of 0.2 for AUC and 16% for accuracy. We show in Figures 14 and 15 the learning performance as well as the confusion matrix of both learning stages of the model. The latter yields an AUC of 0.6333 on the kaggle leaderboard,

making the model less competitive with the previous two approaches. With data augmentation, we managed to improve the performance of the model. Thus, we can say that the model needs more data in order to be able to generalize to unseen data. Therefore, we also applied the semi-supervised approach on MIL ResNet18. Nevertheless, after adding about 250 samples with a confidence of 0.75, we obtained poor performance, equivalent to that of a random classifier. This can be explained by the fact that the base model did not perform well and the added predicted labels were unreliable, so they only biased the model by predicting almost random scores for unseen data, instead of improving its ability to generalize to unknown data.

Appendix

Figure 1: Examples of WSIs of prostate tissue biopsies.

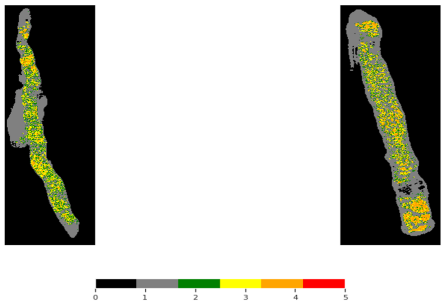

Figure 2: Examples of masks

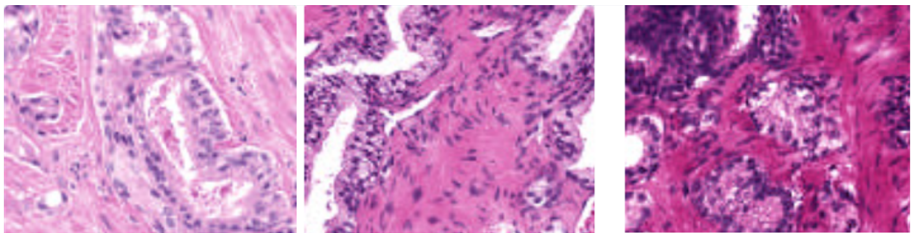

Figure 3: Examples tiles before normalization

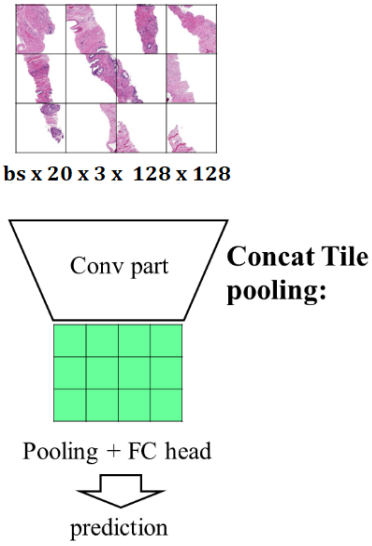

Figure 4: Concatenated Tile Pooling Architecture (4)

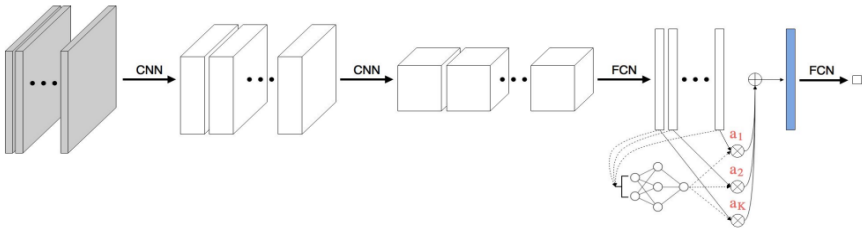

Figure 5: Global architecture of Attention-based deep MIL (Ilse et al., 2018)

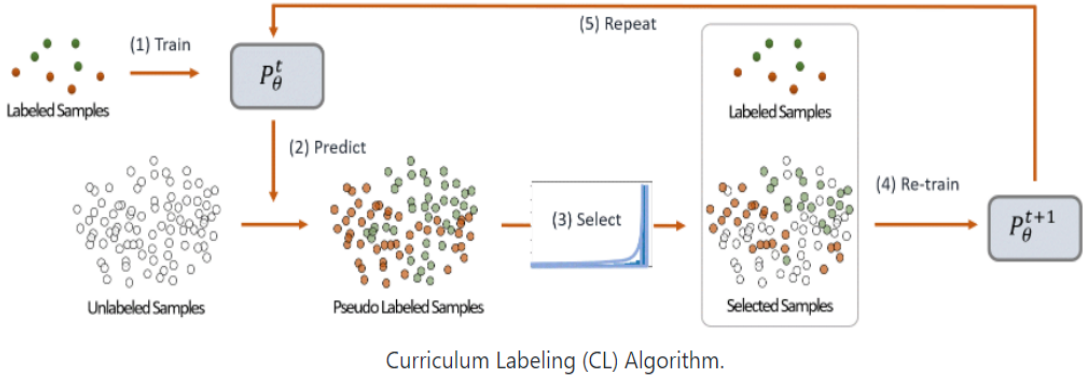

Figure 6: Curriculum Labeling Approach (5)

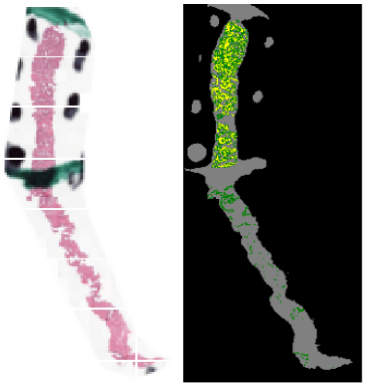

Figure 7: Stained WSI with corresponding mask

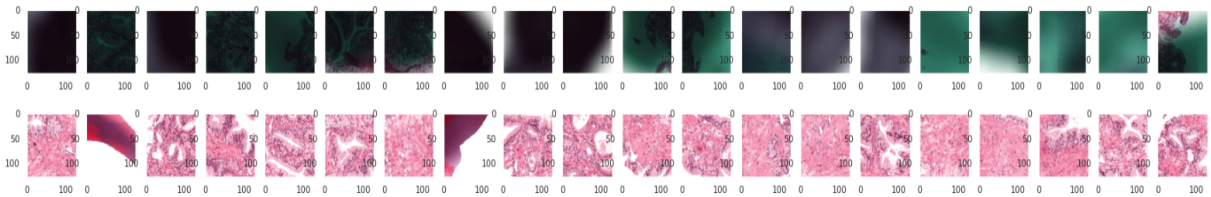

Figure 8: Comparison between two tile-extraction approaches applied on 7.
Up: low-tissue regions with white background.
Buttom: high-tissue regions with black background

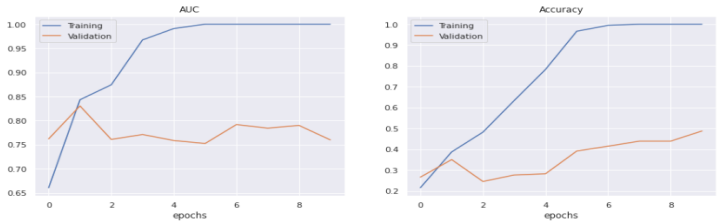

Figure 9: AUC and Accuracy curves in training and validation phase - ResNet18

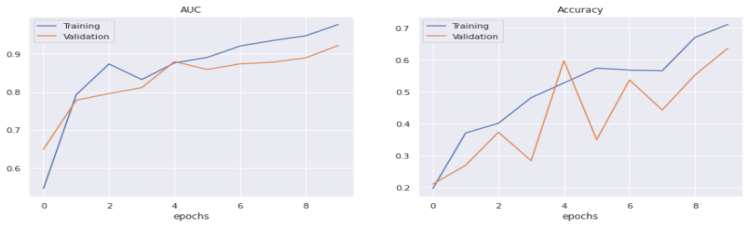

Figure 10: AUC and Accuracy curves in training and validation phase
ResNet18 with data augmentation

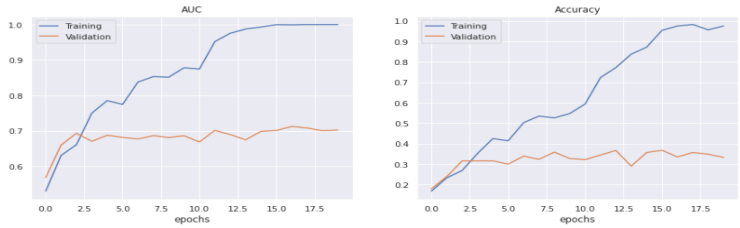

Figure 11: AUC and Accuracy curves in training and validation phase - Pooling ResNet18

Figure 12: AUC and Accuracy curves in training and validation phase
Pooling ResNet18 with data augmentation

Figure 13: AUC and Accuracy curves in training and validation phase - MIL

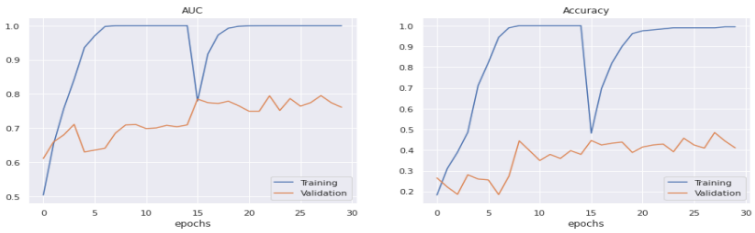

Figure 14: AUC and Accuracy curves in training and validation phase
MIL with data augmentation

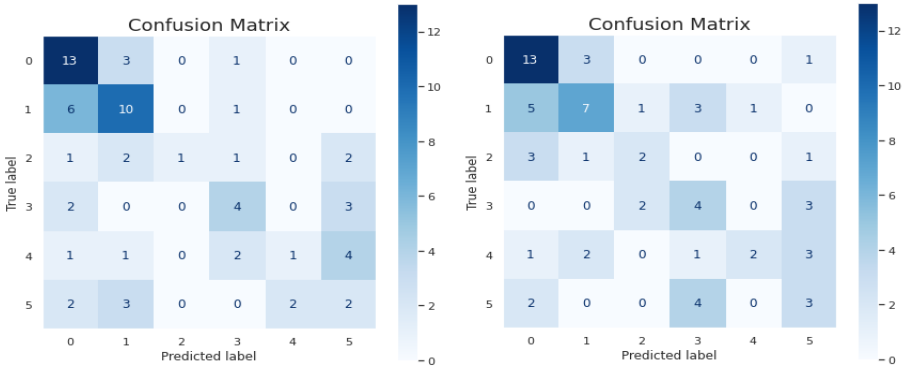

Figure 15: Confusion matrix - MIL with data augmentation.
Left: 1^{st} learning stage. Right: 2^{nd} learning stage

References

https://www.kaggle.com/code/iafoss/panda-16x128x128-tiles/notebook

https://drive.google.com/drive/folders/1GZ8cKy2oUIlJoEvtRpXXaTnHVNbhy2J4?usp=sharing

https://www.kaggle.com/competitions/prostate-cancer-grade-assessment/data

https://www.kaggle.com/code/razamh/panda-concat-tile-pooling-starter-0-79-lb

https://github.com/uvavision/Curriculum-Labeling

Paola Cascante-Bonilla, Fuwen Tan, Yanjun Qi, and Vicente Ordonez. Curriculum labeling: Revisiting pseudo-labeling for semi-supervised learning. *arXiv preprint arXiv:2001.06001*, 2020.

Maximilian Ilse, Jakub Tomczak, and Max Welling. Attention-based deep multiple instance learning. In *International conference on machine learning*, pages 2127–2136. PMLR, 2018.

Jason DM Rennie and Nathan Srebro. Loss functions for preference levels: Regression with discrete ordered labels. In *Proceedings of the IJCAI multidisciplinary workshop on advances in preference handling*, volume 1. Citeseer, 2005.

Abhishek Vahadane, Tingying Peng, Amit Sethi, Shadi Albarqouni, Lichao Wang, Maximilian Baust, Katja Steiger, Anna Melissa Schlitter, Irene Esposito, and Nassir Navab. Structure-preserving color normalization and sparse stain separation for histological images. *IEEE transactions on medical imaging*, 35(8):1962–1971, 2016.