Automated Gleason Grading Using Deep Neural Networks

Gautam Kumar
Parth Dodhia
M.K.Ipsit

June 2020

1 Abstract

Prostrate Cancer is the second leading cause of cancer deaths in men[1]. The Gleason Grading Systems was developed to find the severity of cancer and grade them accordingly based on some specific heterogeneous pattern. This Gleason Grading requires highly trained pathologist. We have designed a automated Annotation system using Deep Learning, where given a WSI(Whole Slide Image) of a Patient, the model predict the type of Gleason Grade. We have trained our system on 641 patients and then evaluated on an independent set of 245 patients. Availability of annotated ground truths enabled us to implement a segmentation model. We also experimented with attention based MIL models on the patch level. We also did not expect a very clear boundary between two different grades of cancer in the same histopathology image, and we were able to predict majority class, that is the type of Gleason Grade reasonably well. Finally, we found that the model performed well in discriminating between Gleason grade 3 and 4.

Contents

2 Introduction

Prostatic Carcinomas are graded according to the Gleason scoring system which was first established by Donald Gleason in 1966[2]. The Gleason Grading System is acknowledged by the World Health Organization(WHO) and has been modified and revised in 2005 and 2014 by the International Society of Urological Pathology(ISUP)[3].. Though there was several changes in the clinical diagnosis methods, Gleason grading remains as one of powerful prognostic tool.The diagnosis using Gleason Grading is based on pattern of tumours. The histological patterns are given different grades between 1 to 5, 1 indicating well differentiated and 5 indicating poorly differentiated. Gleason pattern 4 includes fused glands, cribriform and glomeruloid structures and poorly formed glands. Gleason pattern 5 involves poorly differentiated individual cells, sheets of tumour, solid nests, cords and linear arrays as well as comedonecrosis. Gleason Grade-3 and Gleason Grade-4 are usually present in pairs, and in order avoid diagnosis error and provide correct treatment an automated solution would be extremely useful. In this report, we present an approach using a UNet Model to classify different Gleason Grades of Prostrate Cancer. We have used our evaluation metric as Cohen's Kappa since the test set images were labelled by two pathologists and it also has a class imbalance problem. We also experimented with the recent technique of attention based multiple instance learning.

3 Challenges faced

- Small size of the dataset, which was a larger problem with segmentation since we did not work with patches.
- Uncertainty in ground truths due to inter-pathologist disagreement as seen in the test set (κ score on test set ground truths marked by two different pathologists was 0.44). A few outliers are shown in the figure below :

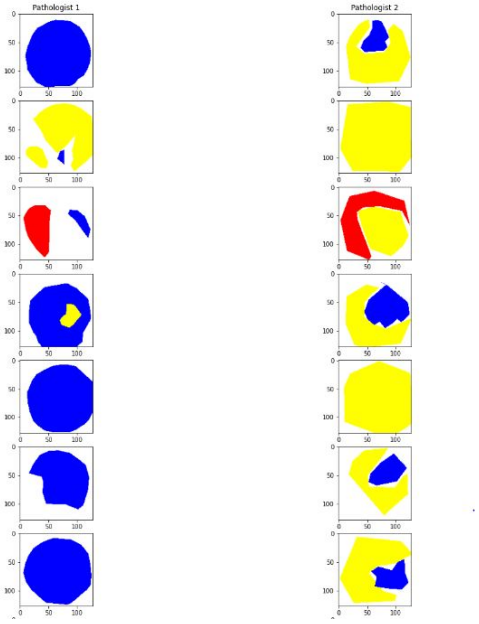

Figure 1: Pathologist 1 vs 2 on the test set

- The classification models we tried were not learning after using attention. The benchmark models like ResNets and MobileNet were giving good results.
- The model was insensitive to both higher and lower learning rates.

4 Motivation Behind the Segmentation Approach

Gleason grading is a multiclass problem and we know exactly which region in the tissue is cancerous since the ground truths are annotated as either benign or with the Gleason grade. So approaching the problem via Segmentation would be interesting and a practical approach. A model performing well in this segmentation task could help in annotation of a larger dataset. We did not expect to achieve sharp boundaries like the ones present in the ground truths, but anticipated that visualisation of the predictions would give us an idea about the different classes present.

5 Datasets and pre-processing

5.1 Dataset

Prostate Cancer is the second leading cause of cancer deaths in men. The Gleason Grading System was developed to find the severity of cancer and grade them accordingly based on some specific heterogeneous pattern. This Gleason Grading requires highly trained pathologist. We have designed an automated annotation system using Deep Learning, where given a WSI(Whole Slide Image) of a Patient, the model predicts the type of Gleason Grade. We have trained our system on 641 patients and then evaluated on an independent set of 245 patients. We decided to use the Harvard Gleason Dataset due to availibility of annotated ground truths for segmentation. We were able to predict the majority class in most of the cases and most of the regions were very clearly segmented. We also did not expect a very clear boundary between two different grades of cancer in the same histopathology image, however we were able to predict majority class, that is the type of Gleason Grade in most of the patient.

5.2 Data Preprocessing

The image size was 3100*3100, and it was infeasible to use directly as it would require a large amount of GPU memory. We resized the images to train on images of size 512*512. We also tried training on smaller images of size 256 but found significant different in results.

6 Segmentation Approach

It took a lot of time and large amount of memory to create sparse or one-hot tensor masks for ground truths separately before starting training, so we implemented this conversion in the getitem function of a custom Dataset class while training itself (using Dataloaders we were able to almost completely eliminate the time required for this)
We resized the 3100 sized whole slide images to a size of 512 for training. The input images were normalised as per the ImageNet mean and variance. We used data augmentations like random horizontal and vertical flips to increase the dataset size, and color jitter while training.

7 Experiments and Results

7.1 Network Architecture

We experimented with the standard UNet (symmetric encoder decoder structure) architectures. Up-sampling in the decoder was done using transposed convolutions. We found pre-trained encoders to work much better than starting from scratch (decoders were randomly initialised). First we tried using a VGG-11 pre-trained encoder, but the training process was quite slow and the model occupied a lot of GPU memory as well. Finally we chose a pretrained ResNet34 encoder with the encoder decoder connections after each res-block. Dropout was also added after each encoder and decoder layer to tackle overfitting.

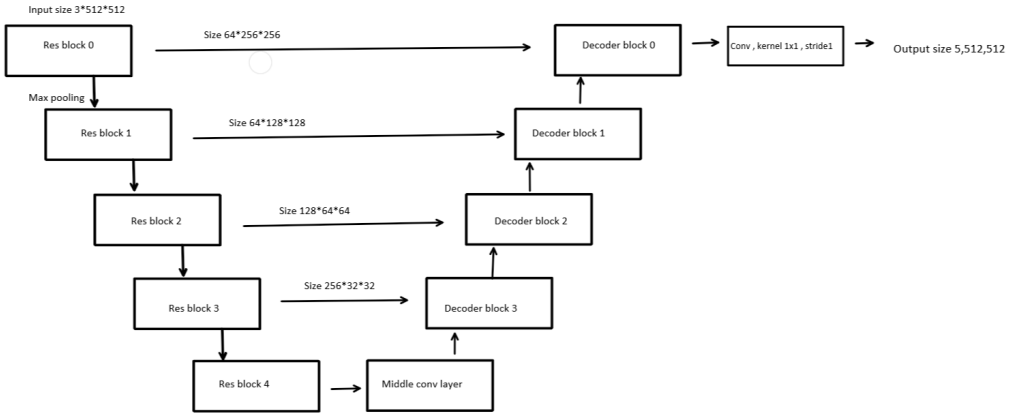

Figure 2: UNet with Resnet 34 encoder

Resblock 0 consists of the first convolutional layer of kernel size 7 in Resnet34 (which is followed by the maxpool layer shown explicitly). Subsequent Resblocks are all the convolutional layers in Resnet34 with same number of channels stacked together.

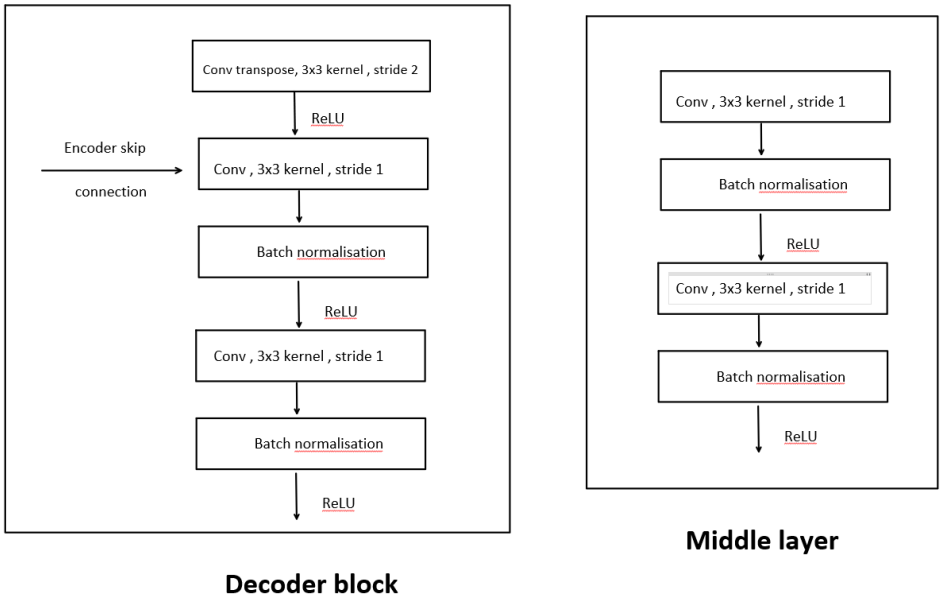

Figure 3: Middle layer and Decoder blocks

7.2 Training

We worked on batches of size 8 and found SGD to work equally well compared to Adam. We used SGD optimiser with momentum and a reduce learning rate on plateau (validation loss) scheduler. We tuned the learning rate, dropout probability (constant for all layers) and gamma for focal loss, and found that cross entropy loss (gamma=0) worked equally well as compared to focal loss for higher gamma values. We found a high learning rate to be effective.
Finally, we found better performance with a combination of cross entropy and soft dice loss (arbitrary, nearly equal weights for both). We monitored the loss and accuracy curves and periodic predictions (every 15-20 epochs) on tensorboard during training.

7.3 Loss Functions

Cross-Entropy Loss

Multi-class crossentropy loss is given by the following formula

$$\mathscr{L}_{CE} = -\sum_{c=1}^{M} y_c \log p_c$$

where y_c is the label and p_c is the predicted probability and M is the total number of classes [6].

Soft Dice Loss

It is based on intersection over union for two images. This is calculated for an entire image.

$$\mathscr{L}_{Dice}(p,q) = 1 - \frac{1}{M}\sum_{c=1}^{M} \frac{2 \times \sum_{i,j} p_{cij} q_{cij} + \varepsilon}{\left(\sum_{i,j} p_{cij}^2\right) + \left(\sum_{i,j} q_{cij}^2\right) + \varepsilon}$$

where p_{cij}, q_{cij} are the prediction and label for the pixel (i,j), M is the total number of classes, ε is a small positive number for numeric stability[8].

Focal Loss

Multi-class focal loss, which is a modification of crossentropy loss is given by the following formula [7]

$$\mathscr{L}_{Focal} = -\sum_{c=1}^{M} y_c (1-p_c)^{\gamma} \log(p_c)$$

where y_c is the label, p_c is the predicted probability and M is the total number of classes. $\gamma \geq 0$ is a tunable focusing parameter with values generally in the range $[0,5]$.

7.4 EVALUATION CRITERIA

Cohen's Kappa
Cohen's kappa coefficient is a statistic that is used to measure inter-rater reliability (and also Intra-rater reliability) for qualitative (categorical) items It is generally thought to be a more robust measure than simple percent agreement calculation, as takes into account the possibility of the agreement occurring by chance. As the data is labelled by two pathologist this is good metric for measurement.

7.5 RESULTS

Table 1: Results on the Test Dataset

Cohen's Kappa Score between Pathologists	0.44
Cohen's Kappa Score between Model and Pathologist I	0.53

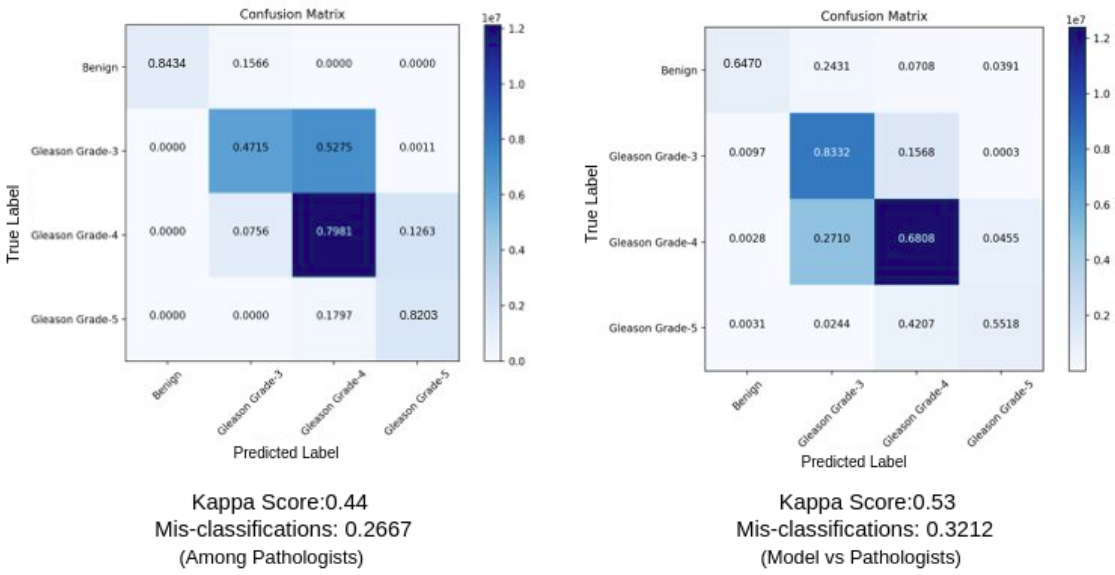

Figure 4: Confusion Matrix

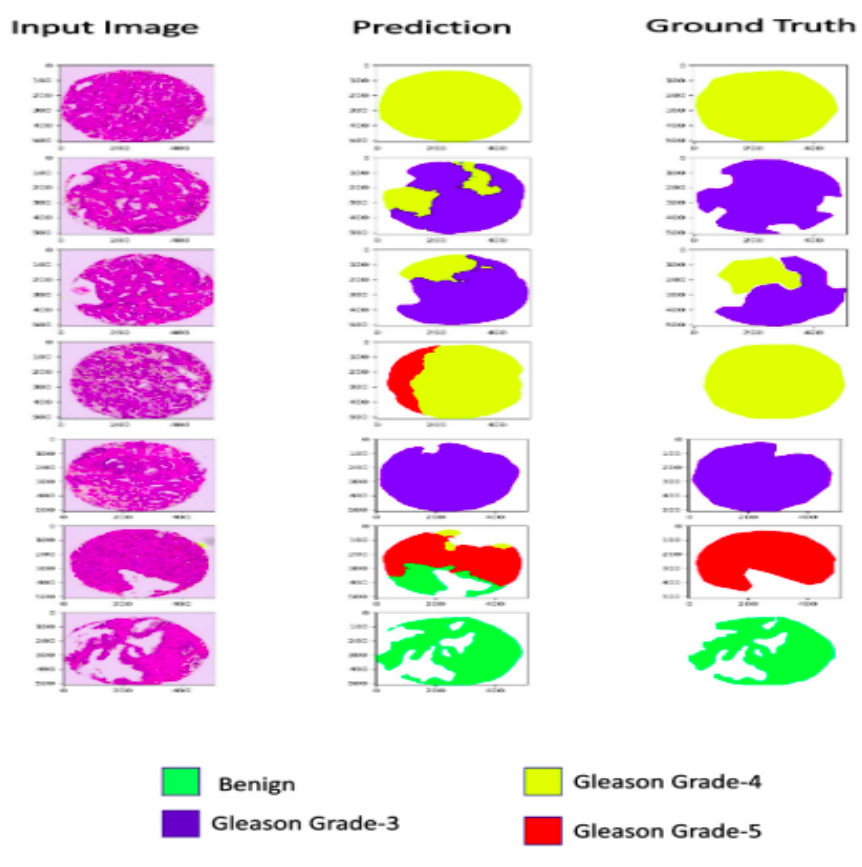

Figure 5: Visualization of Segmentation Results

.

8 Discussion and Conclusion

The segmentation approach gave reasonable results with good performance on the test as compared to agreement between pathologists. It did quite well in discriminating between Gleason grades 3 and 4 which is an important factor for diagnosis. Future work could be to feed a multi-resolution input to the UNet for better performance.

9 Attention Based Multiple Instance Learning

9.1 Motivation

As the name suggests, attention based multiple instance learning is an architecture that learns features based on the instances of its occurence. The logic is that using A-MIL, more robust features can be learnt. MIL provided good results both on normal object classification and on some of the histopathology datasets. We plan to use try this approach to detect Prostate Cancer.

9.2 Data Preprocessing

In our first approach, we used the entire data and attempted multi-class classification using A-MIL. We divided the data into 10 classes considering all the combinations of gleason scores into account. We made bags for each WSI of size 3100×3100 by dividing it into smaller patches of size 750×750. The label for the bag was given based on the gleason score of that WSI. We also experimented with different patch sizes of $31 \times 31, 62 \times 62, 155 \times 155$.
For our second approach, from the entire, we made patches of size 1024×1024 from each image with a condition that the entire patch has both the primary and secondary gleason score same. This gave us 4 classes, benign, Gleason Grade3, Gleason Grade 4 and Gleason Grade 5.

We further divided each patch into 5 smaller patches of size 500×500 and formed a bag. The label of the bag is same as the label of the parent patch. Making of bag was done during training and not during data preparation because of the framework limitation. We used `torchvision.datasets.ImageFolder` function and a dataloader to set up the data efficiently for training [10]. Data augmentations like random horizontal and vertical flipping, and normalizing using ImageNet statistics were used.

9.3 Approach

In our first approach, we made our model based on the Attention Based Deep Multiple Instance Learning paper. We used LeNet as feature extractor. We then tried both attention and gated attention with learnable MIL Pooling using dense layers. After the MIL pooling, we used dense layers follwed by a softmax layer at the end for classification. Later, we also tried using MobileNetv2 and ResNet18 as feature

extractors. In our second approach, we used ResNet18 as the feature extractor and then used attention followed by a dense classifier block with a softmax layer in the end.

9.4 Architecture

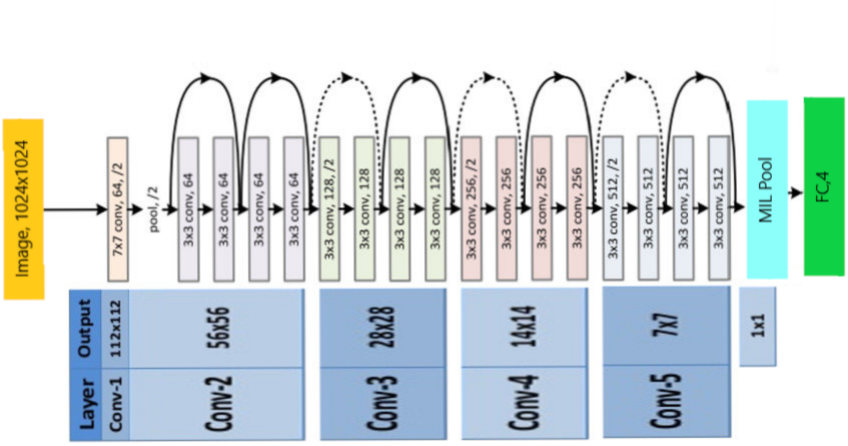

Figure 6: A-MIL Model

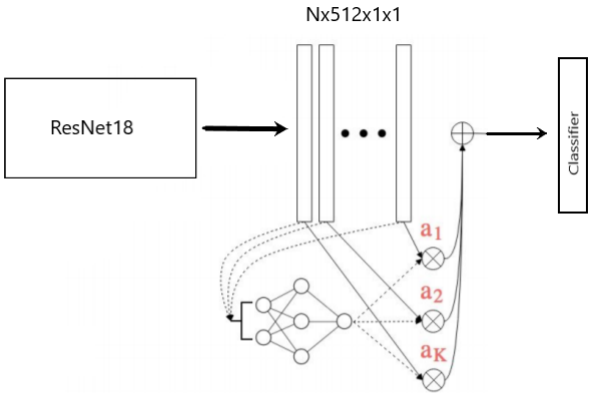

Figure 7: MIL Pooling

9.5 Training

We used a batch size of 1 and before passing the image into the model, we made a bag of sub-patches. This was passed as input to the model. There was a single output for each bag.

In our first approach, we used crossentropy loss and Adam optimizer with different learning rates. We used different learning rate schedulers like StepLR, ReduceLROnPlateau, and Cyclic LR. But the model was performing not good. The prediction for every slide was the same. This was not of much help.

In our second approach, we first finetuned ResNet18 with the data from pure patches. The last fc layer had 4 classes. So, we removed the last FC layer and used the remaining network as a feature extractor for the embedding based attention model. We then found a peculiar behaviour.

We also tried block-wise training. First we trained ResNet18 as a feature extractor, froze it weights, and then trained the attention block seperately. This also was not giving any good results.

9.6 Loss Function

Cross-Entropy Loss

Multi-class crossentropy loss is given by the following formula

$$\mathscr{L}_{CE} = -\sum_{c=1}^{M} y_c \log p_c$$

where y_c is the label and p_c is the predicted probability and M is the total number of classes.

.

9.7 MIL Pooling

The technique that we used to pool lower dimensional embeddings of inputs, which is independent of the size of the bag is shown below. Let $H = \{h_1, h_2, \ldots h_K\}$ be a bag of K embeddings, then the proposed pooling is

$$
\begin{aligned}
z &= \sum_{k=1}^{K} a_k h_k \\
&\text{where} \\
a_k &= \frac{exp(w^T \tanh(V h_k^T))}{\sum_{j=1}^{K} exp(w^T \tanh(V h_j^T))}
\end{aligned}
$$

where $w \in \mathbf{R}^{L \times 1}$ and $V \in \mathbf{R}^{L}$ are parameters. This has been realized using dense neural networks [9].

9.8 Evaluation Metrics

We used patch-level classification accuracy as the evaluation metric. The best accuracy was approximately 75% without attention and 52.4% with attention. The details are as follows:

Metric	Blue	Green	Red	Yellow
Precision	**0.5748**	0.144	0.235	**0.59**
Recall	**0.5995**	0.3272	0.0615	**0.5402**
F1 Score	**0.5869**	0.2	0.097	**0.564**

Table 2: Results on the Test Data

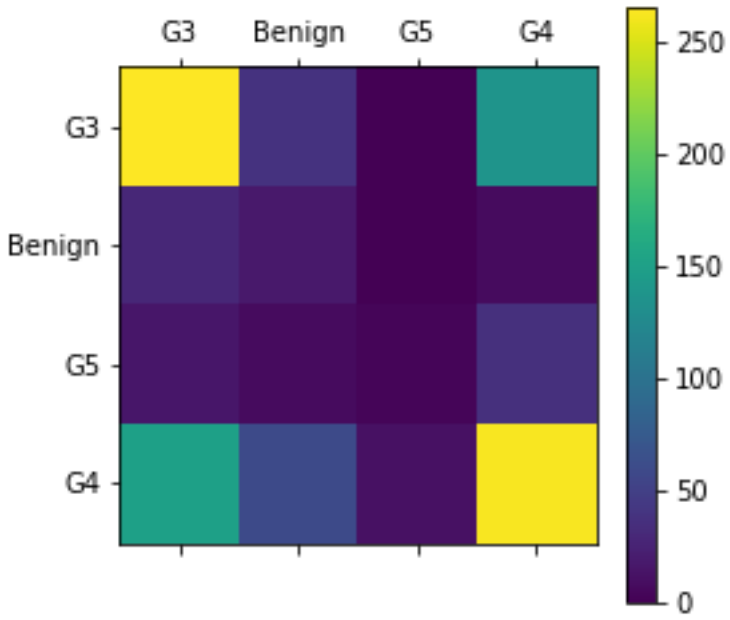

Figure 8: Confusion Matrix

9.9 Discussion and Conclusion

Attention based classification was not working for all the classes. The model is easily able to identify Gleason 3 and Gleason 4. The training process was very unstable. We tried various schedulers and increased the depth of the attention and classification layers, but we couldn't get good results. The model was overfitting the train set with train accuracies reaching more than 80%.

Further work can be to use segmentation with self guided attention as mentioned in the given paper.

References

[1] WHO Classification of Tumours of the Urinary System and Male Genital Organs. International Agency for Research on Cancer (IARC) (2016)

[2] Gleason, D. F. Mellinger, G. T. Prediction of prognosis for prostatic adenocarcinoma by combined histological grading and clinical staging. J. Urol. 111, 58–64 (1974)

[3] Faraj, S. F. et al. Clinical Validation of the 2005 ISUP Gleason Grading System in a Cohort of Intermediate and High Risk Men Undergoing Radical Prostatectomy. PLoS One11, e0146189 (2016).

[4] Gordetsky, J. Epstein, J. Grading of prostatic adenocarcinoma: current state and prognostic implications. Diagn. Pathol.11, 25 (2016).

[5] Epstein, J. I. Prostate cancer grading: a decade after the 2005 modified system. Mod. Pathol.31, S47–63 (2018)

[6] https://en.wikipedia.org/wiki/Cross_entropy

[7] https://towardsdatascience.com/review-retinanet-focal-loss-object-detection-38fba6afabe4

[8] https://arxiv.org/abs/1707.03237 Generalised Dice overlap as a deep learning loss function for highly unbalanced segmentations.

[9] https://arxiv.org/pdf/1802.04712.pdf Attention-based Deep Multiple Instance Learning

[10] https://pytorch.org/docs/stable/torchvision/index.html