
Lymphoma Subtype Diagnosis with Deep Learning

Vrishab Krishna
Stanford University
vrishab@stanford.edu

Abstract

Lymphoma is the broad term for cancers arising in the lymphatic system. Subtype prediction for lymphoma is crucial for oncologists to determine treatment regimes. In this work, we apply transfer learning to a cohort of lymphoma subtype cases with high resolution tissue micro-arrays from Guatemala to narrow the list of potential diagnoses and enable definitive diagnosis of lymphoma. Due to the lack of patch-level annotations, we experiment with standard supervised learning and multiple instance learning while testing different loss functions to deal with label imbalance. This system has the potential to reduce the requirement of ancillary studies and help pathologists in low-income environments to make more accurate diagnoses.

1 Introduction

Lymphomas are neoplasms derived from lymphocytes and vary from indolent lymphomas that can be managed with a "watch and wait" approach to aggressive types that require prompt administration of high intensity chemotherapy. The accurate diagnosis of lymphomas is essential in order to properly guide patient management and treatment. This requires morphologic evaluation of hematoxylin and eosin (H&E)-stained tissue by a trained pathologist, who then orders additional ancillary studies such as immunohistochemical stains to arrive at a definitive diagnosis. Treatment decisions are strongly influenced by the particular subtype of lymphoma (primarily based on cell of origin) which is suggested by the histologic features on H&E-stained slides and confirmed by either immunohistochemistry, flow cytometry, or both. Thus, the diagnosis of lymphoma subtypes frequently requires extensive immunophenotypic studies (by immunohistochemistry or flow cytometry).

Worldwide, there are not enough pathologists to meet the needs of patients and the immunohistochemical stains and other ancillary studies (e.g. fluorescence in situ hybridization) are expensive and frequently not available in less-developed countries. Deep learning and computer vision solutions have been shown to perform as well as pathologists in certain tasks and thus represent promising tools that may help bring diagnostic capacity to underserved areas of the world [12, 20].

Through this project, we aim to develop a computational system which predicts the particular lymphoma subtype from Tissue Micro-Arrays (TMAs). The final purpose of this endeavour would be to have a system that could narrow down the list of potential subtypes for a pathologist to make the final diagnosis. This would potentially reduce the number of immunohistochemical stains needed to arrive at a definitive diagnosis and improve the efficiency of diagnosis in resource-rich and resource limited settings.

2 Related Work

Deep learning applied to digital pathology is a growing field of research with a number of open datasets [4, 10] with models trained on different tasks like tumour localization, nuclei prediction and classification [8], tissue classification, cellularity prediction, etc. Self-supervised models trained on many datasets with different tasks have also been found to achieve high accuracy with transfer learning and fine-tuning [7, 17].

Due to the time intensive nature of labelling digital pathology data, methods to use smaller amounts of labelling are widespread. Generally, pathologists annotate small patches of a much larger slide image to tell if each individual patch has a particular form of cancer or not. Now, methods using weakly supervised learning are used to train models with a single label on group of patches [2] or a single label for the entire slide [5, 14]. Since we do not have labels for each patch, we utilize methods from these papers for training.

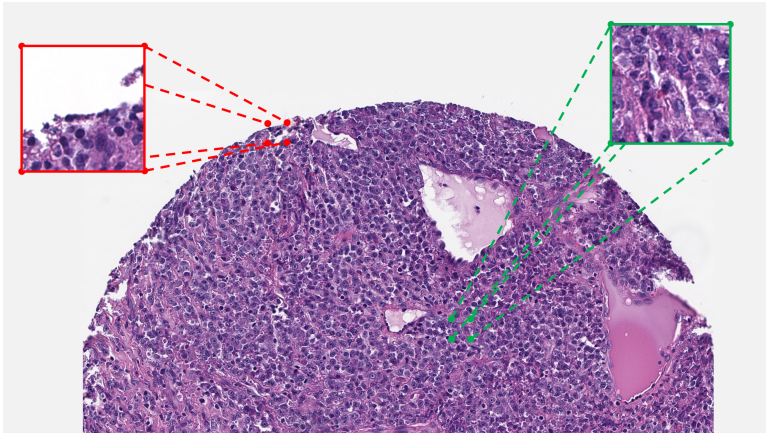


Figure 1: Patch extraction. The patch on the right is extracted whereas the patch on the left is filtered out due to the lower amount of tissue in the image (thresholding by saturation)

Thus far, deep learning tools to make lymphoma diagnoses have focused on the diagnosis of a small number of lymphoma subtypes with supervised, pre-trained CNN architectures. Li et al. proposed an ensemble of 17 different pre-trained CNNs to distinguish between DLBCL and non-DLBCL [12]. In another study, Zhang et al. proposed a diagnostic method to distinguish three major NHL subtypes: chronic lymphocytic leukemia, follicular lymphoma, and mantle cell lymphoma using transfer learning with fine-tuned VGG, ResNet, and DenseNet variants and principal component analysis (PCA) [20]. These methods focus on predicting a small set of subtypes, not sufficient for clinical use. We are focusing on clinically relevant bins as part of our dataset which could provide greater benefit to pathologists.

One interesting fact to note is that a large number of these datasets are based on Whole Slide Images (WSIs). TMAs are developed by extracting regions of interest from the WSIs which are enriched for cancerous cells. To the best of our knowledge, no studies focus on using TMAs for lymphoma subtype prediction with deep learning. This study represents an important advance in the application of deep learning tools to lymphoma subtype diagnosis.

3 Data Processing

3.1 Main Dataset and Preprocessing

The dataset was collected as part of the validation transcriptional profiling method to predict lymphoma subtypes [19]. Whole slide samples collected from Guatemala were processed into Tissue Micro-Arrays (TMAs) and then scanned to uint8 RGB SVS files. The hematopathologist’s prediction on the basis of immunohistochemical stains is the gold standard for lymphoma diagnosis. These predictions are binned into 9 clinically significant groups as the labels for our task.

The image data consists of 8 Tissue Micro-Arrays (TMAs) with samples from a total of 654 patients in Guatemala provided to us by Stanford Medicine. Each sample is called a core (≈ 900 cores in the dataset) and contain hundreds of cells corresponding to a single patient. The TMAs are stored as SVS files with dimensions $\approx 50000 \times 50000$ pixels at a $40\times$ magnification. The cores are labeled and annotated using *QuPath*, an open-source tool for digital pathology [3].

Due to the high resolution of such files, models are trained on square patches extracted from each core. The *QuPath* annotations are used to create bounding boxes around each core. Using the *OpenSlide* package in Python, we extracted 224×224 pixels patches of the region within the box [16]. These are then filtered using the average saturation (from the HSV colorspace) of the image to prevent backgrounds of the slide to enter into the dataset. We find that, in general, a core has ≈ 100 patches of dimension 224×224 . This can be increased with augmentations like overlapping patches during extraction: we apply a 50% overlap in extraction.

Most current research on lymphoma subtypes focuses on images at a high magnification so patches were obtained at the level of $40\times$ magnification. Given 900 TMAs, this constitutes around 100k images. Hence, we store the patches in HDF5 format for speed of file I/O [18].

3.2 Challenges

Note that as cancerous cells might be only present at a particular part of the core, each of the patches could have a different label than the label of the overall core. We do not have a well-defined label for each patch. This poses a challenge in the training approach. We describe two different training methodologies to allow for this flexibility.

An issue with the current dataset is that there is a large skew in cores corresponding to a particular label. For example, the most common form of lymphoma, Diffuse Large B-cell Lymphoma (DLBCL) accounts for 30% of all lymphoma cases but accounts

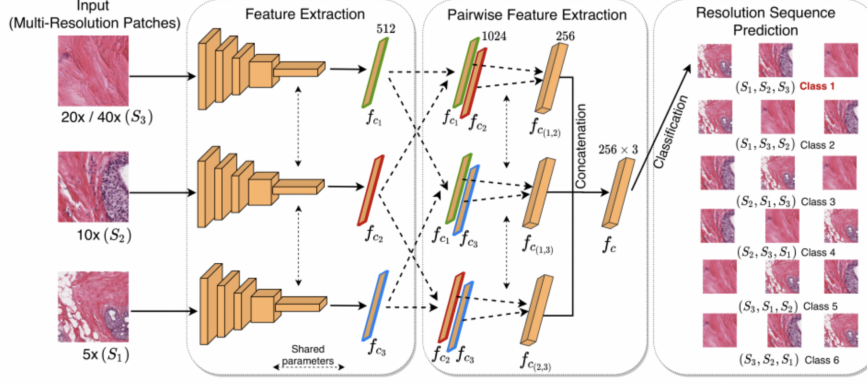


Figure 2: TripletNet Architecture from Srinidhi et. al. This is the Resolution Sequence Prediction (RSP) pretext task used for self-supervised representation learning. During supervised training, we pass the same image as input to the base CNNs and use the 256×3 representation f_c as input to a linear layer for the final prediction.

for 50% of the labels in this dataset. Some other lymphoma subtypes have only 5-6 cores. We take measures to counteract this dataset imbalance with different loss functions and .

4 Methods

4.1 TripletNet Architecture

We make use of the TripletNet architecture for deep metric learning [9]. Srinidhi et. al. used TripletNet pretraining to overcome the need for a sizeable number of labelled instances. This was achieved using “by leveraging both task-agnostic and task-specific unlabeled data” [17]. They used two strategies: a self-supervised pretext task on different resolution of histology images and a teacher-student paradigm to transfer representations to downstream tasks. We chose this model as it performed well on multiple histopathology benchmarks such as Camelyon16 (tumor metastasis detection), BreastPathQ (tumour cellularity quantification), and the Kather Multiclass dataset (tissue type classification). Since we train model on H&E patches, we apply transfer learning by using the weights trained on the Camelyon16 task [4] which also consists of H&E-stained slides. The TripletNet base model is applied to each patch to obtain a $256 \times 3 = 768$ length feature vector (f_c in the figure). This feature vector is used as input to a linear layer with 9 logits as output for the 9-way classification.

4.2 Learning Regimes

4.2.1 Multiple Instance Learning

Although we have a very large number of images, each of these images does not have a clear label. Since a given patch of the above dimensions contains at maximum 50 cells and is only a very small part of the whole core, it might not contain any cancerous cells at all. Hence, the actual label of the patch might be “non-cancerous” but if we simply use the core label, we would get an incorrect labelling. A common method used to solve this challenge in histology image classification is Multiple Instance Learning (MIL) [6]. In MIL, we train the model on a “bag” of instances which, when taken together, have a particular label. This is done by aggregating representations/predictions to obtain a final prediction for the bag on which the loss is calculated.

In this task, if any one of the patches are predicted as positive, the whole bag would be predicted as positive. We use the TripletNet representations followed by a linear layer to obtain the class logits. We apply maxpooling across the logits for different patches to obtain a final set of bag logits which, after a softmax, give the final predictions for the bag.

4.2.2 Naive/Standard Supervised Training

One issue with MIL is that for each backpropagation, we need to compute the results for each member of the bag, i.e., the number of forward propagations is equal to the bag size. This means that training is very slow using MIL. To combat this, we also experiment with the naive assumption: the label of each patch is the label of its corresponding core. An interesting aspect of using TMAs is that the cells are highly enriched for lymphoma - potentially 30 – 60% of the cells in the cores are cancerous. Hence, we could still use the standard supervised training regime for a classification task without any significant loss in accuracy. We refer to this training procedure as the “Naive” training regime in the rest of the paper.

Training Regime	Loss Type	Accuracy		
		Train (%)	Val (%)	Test (%)
<i>MIL</i>	<i>Weighted CE</i>	<i>49.0</i>	<i>39.9</i>	<i>39.6</i>
MIL	Focal	48.0	39.9	39.6
Naive	Weighted CE	47.3	42.7	38.0
Naive	Focal	47.3	42.7	38.0
Naive (E2E)	Weighted CE	47.3	42.7	38.0
Naive (E2E)	Focal	54.4	41.3	41.4
Unspecialized Pathologist				56.7

Table 1: Results for 9-way classification for different training regimes and loss functions. Bold represents the baseline model. Note that the accuracy for the naive regime is the patch level accuracy while that for the MIL regime is the core level accuracy.

4.3 Training and Evaluation

The model development and training was performed with *Pytorch* and *Pytorch Lightning* on the SAIL Cluster at Stanford [1, 15]. For training, we use the Adam optimizer with $\beta_1 = 0.9, \beta_2 = 0.999$ [11]. Hyperparameter tuning was done exhaustively over an exponential interval with 4 possibilities ($10^{-2}, 10^{-3}, 10^{-4}, 10^{-5}$). In naive training, we consider 3 possible batch sizes (8, 32, 64). The bag size in MIL was set to 1 due to computational constraints while the bag size was set to 64. The 64 patches were randomly sampled from the ≈ 100 patches extracted from each core. In cases where the number of patches was fewer, the bag size is reduced to the maximum number of patches available.

We experiment with feature extraction and end-to-end training. For feature extraction, representation vectors were extracted for each patch and stored in the HDF5 format [18]. Due to computational constraints, we could only run end-to-end training with the Naive regime and, even then, only for a few epochs.

During initial experiments, we trained the model on categorical crossentropy (CE) but obtained poor results due to class imbalance. To account for this, we test two different loss functions that take into account the skew of labels towards a few classes: the weighted crossentropy (weighted CE) loss and the focal loss. We weight the crossentropy loss based on the inverse proportion of each class in the training set (see Appendix 8.2.1 for mathematical details). For focal loss, we set the hyperparameters $\gamma = 2.0$ and α is either 0.05, 0.1, 0.15, 0.2, 0.25, smaller for the more highly represented classes and larger for the sparse classes (see Appendix 8.2.2 for more details).

We test the weighted crossentropy loss and focal loss by training for 50 epochs and choosing the model with the best validation accuracy. The train-val-test split was 80:10:10 across the 654 patients such that patches from the same patient were not shared across split. For each model, we report accuracy as the primary metric and use confusion matrices for more detailed analysis of results. Our standard of human performance on this task is the performance of an unspecialised pathologist using only H&E slides to make a diagnosis (i.e. not using the gold standard of immunohistochemical stains).

5 Results and Discussion

Our initial experiments (for the milestone) were focused on the MIL regime. Our baseline model with weighted CE obtained a test accuracy of 39.61% (Table 5) and had the confusion matrix shown in Figure 3. From the figure, we see that the model was almost always predicting the most prevalent class of DLBCL in spite of weighting the loss function.

We theorized that this could be due to three main reasons. The first is that the MIL regime requires too many computations for each backpropagation so the model was not updated enough or gradients were not (a batch size of 1 was the maximum possible due to computation constraints). To test this, we experimented with the naive training method to deal with computational limitations. The second reason is that the weighted crossentropy function was not sufficient to deal with the class imbalance. To reduce this effect, we test the focal loss function with both MIL and naive regimes. The final issue is that the baseline model consisted of just replacing the projection head with a linear layer to 9 logits rather than training than finetuning the whole TripletNet model. Due to computational difficulties, we only experiment with training the end-to-end model with the naive regime.

5.1 Experiments on Regimes and Losses with Linear Layer

We first focus on the task with training a linear layer on the TripletNet features. The highest accuracy was obtained with a learning rate of 10^{-3} and batch size of 64 for the naive regime. The results for all these tasks can be found in the first four rows of Table 5. Note that irrespective the loss, for the feature extraction tasks the accuracies are the same for each regime. When we observe the corresponding confusion matrices (see Appendix Figure 6 and Figure 7), we see that the issue of label imbalance still adversely affects the model’s predictions, causing it to almost always DLBCL. This is not solved by changing

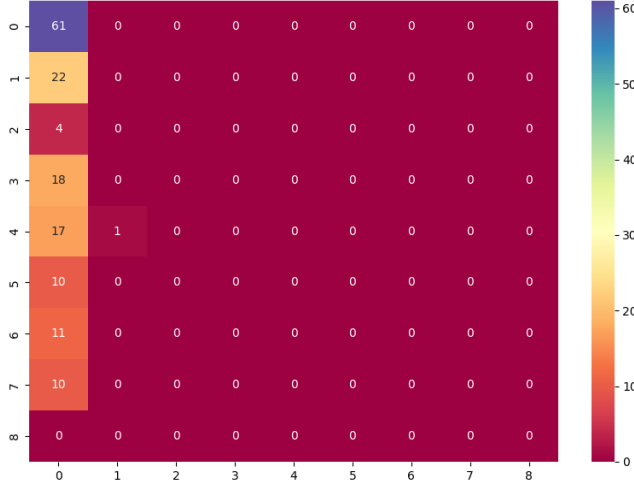


Figure 3: Confusion matrix for baseline MIL model with weighted crossentropy loss (on 50 epochs)

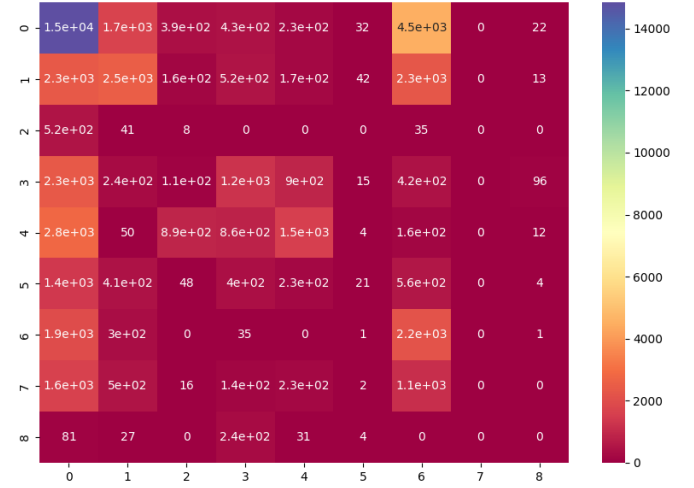


Figure 4: Confusion matrix for End-to-end Naive model with focal loss (on 4 epochs)

loss functions or with the naive regime. We feel that this is a case of high bias and the model is not capable of learning the differences between the classes with just a linear layer over the extracted features.

5.2 Naive Regime with End-to-end Training

We use the same hyperparameters as the previous experiments. While we could not train the end-to-end model for a large number of epochs (only 4), it obtained much better results than the linear layer. While the actual metrics look very similar in Table 5, the confusion matrix in Figure 4 illustrates a very different result.

The end-to-end model trained on weighted crossentropy suffered adversely due to class imbalance and had a almost identical accuracies (see Table 5) and confusion matrix (like Appendix Figure 6). The model trained with focal loss was the only model that made predictions across all the labels in the test set and its confusion matrix (Figure 4) has strong predictions across the diagonal for most classes. However, the model does overfit on the training data, maybe necessitating more augmentations or using dropout or weight decay.

While a large number of the incorrect predictions could be reduced by further training, errors in classes like MZL (class 5), FL (class 3), etc. could be due to the lack of large scale texture features generally observed by pathologists on low magnification WSIs. These features are not present in TMAs. Another aspect to consider is that some of the patches which do not contain cancer would be incorrectly labeled with a particular subtype of cancer in the naive regime. This could be a potential reason for why a large number of patches are predicted incorrectly across the training and validation set as well (see Appendix Figure 8, 9).

In all the cases, accuracy is close to the performance of the unspecialised pathologist (56.7%). However, there is still significant room for improvement.

6 Conclusions and Future Work

Through this work, we developed a framework for predicting lymphoma subtypes from TMAs. We developed a pipeline for preprocessing, augmenting, and extracting patches from high resolution TMA cores. We experimented with MIL and naive training regimes subject to computational constraints and obtained encouraging results. This supports the claim that TMAs can be used to predict lymphoma subtypes and gives encouraging performance on classifying across largest number of clinically significant lymphoma subtypes with deep learning. However, in comparison with human performance there is still room for improvement. With enough computational resources, we hope to experiment with end-to-end MIL training as well as a more comprehensive set of pretrained architectures. In addition, we could attempt to get the benefits of both regimes by pretraining the model with the naive regime and train a projection head with the MIL regime to obtain core level classifications.

7 Acknowledgements

I am working on this project as part of the AI for Healthcare Bootcamp with Vivek Shankar and Xiaoli Yang under Dr. Pranav Rajpurkar of the Stanford ML Group and in collaboration with Dr. Sebastian Fernandez-Pol of the Department of Pathology at Stanford. I would like to thank all of them and TA Ruta Joshi for her valuable advice on the project and this report.

References

- [1] Pytorch lightning: The lightweight pytorch wrapper for high-performance ai research.
- [2] H. E. Achi, T. Belousova, L. Chen, A. Wahed, I. Wang, Z. Hu, Z. Kanaan, A. Rios, and A. N. D. Nguyen. Automated Diagnosis of Lymphoma with Digital Pathology Images Using Deep Learning. *Ann Clin Lab Sci*, 49(2):153–160, Mar 2019.
- [3] Peter Bankhead, Maurice B. Loughrey, José A. Fernández, Yvonne Dombrowski, Darragh G. McArt, Philip D. Dunne, Stephen McQuaid, Ronan T. Gray, Liam J. Murray, Helen G. Coleman, Jacqueline A. James, Manuel Salto-Tellez, and Peter W. Hamilton. QuPath: Open source software for digital pathology image analysis. 7(1), December 2017.
- [4] Babak Ehteshami Bejnordi, Mitko Veta, Paul Johannes van Diest, Bram van Ginneken, Nico Karssemeijer, Geert Litjens, Jeroen A. W. M. van der Laak, Meyke Hermesen, Quirine F. Manson, Maschenka Balkenhol, Oscar Geessink, Nikolaos Stathonikos, Marcory CRF van Dijk, Peter Bult, Francisco Beca, Andrew H. Beck, Dayong Wang, Aditya Khosla, Rishab Gargeya, Humayun Irshad, Aoxiao Zhong, Qi Dou, Quanzheng Li, Hao Chen, Huang-Jing Lin, Pheng-Ann Heng, Christian Haß, Elia Bruni, Quincy Wong, Ugur Halici, Mustafa Ümit Öner, Rengul Cetin-Atalay, Matt Berseth, Vitali Khvatkov, Alexei Vylegzhanin, Oren Kraus, Muhammad Shaban, Nasir Rajpoot, Ruqayya Awan, Korsuk Sirinukunwattana, Talha Qaiser, Yee-Wah Tsang, David Tellez, Jonas Annuscheit, Peter Hufnagl, Mira Valkonen, Kimmo Kartasalo, Leena Lanttonen, Pekka Ruusuvaari, Kaisa Liimatainen, Shadi Albarqouni, Bharti Mungal, Ami George, Stefanie Demirci, Nassir Navab, Seiryō Watanabe, Shigeto Seno, Yoichi Takenaka, Hideo Matsuda, Hady Ahmady Phoulady, Vassili Kovalev, Alexander Kalinovsky, Vitali Liatuchuk, Gloria Bueno, M. Milagro Fernandez-Carrobles, Ismael Serrano, Oscar Deniz, Daniel Racoceanu, and Rui Venâncio and. Diagnostic assessment of deep learning algorithms for detection of lymph node metastases in women with breast cancer. *JAMA*, 318(22):2199, December 2017.
- [5] Gabriele Campanella, Matthew G. Hanna, Luke Geneslaw, Allen Mirafior, Vitor Werneck Krauss Silva, Klaus J. Busam, Edi Brogi, Victor E. Reuter, David S. Klimstra, and Thomas J. Fuchs. Clinical-grade computational pathology using weakly supervised deep learning on whole slide images. *Nature Medicine*, 25(8):1301–1309, July 2019.
- [6] Marc-André Carbonneau, Veronika Cheplygina, Eric Granger, and Ghyslaine Gagnon. Multiple instance learning: A survey of problem characteristics and applications. *CoRR*, abs/1612.03365, 2016.
- [7] Ozan Ciga, Tony Xu, and Anne L. Martel. Self supervised contrastive learning for digital histopathology, 2021.
- [8] Simon Graham, Quoc Dang Vu, Shan E. Ahmed Raza, Ayesha Azam, Yee Wah Tsang, Jin Tae Kwak, and Nasir Rajpoot. Hover-net: Simultaneous segmentation and classification of nuclei in multi-tissue histology images. 58:101563, December 2019.
- [9] Elad Hoffer and Nir Ailon. Deep metric learning using triplet network, 2018.
- [10] Andrew Janowczyk and Anant Madabhushi. Deep learning for digital pathology image analysis: A comprehensive tutorial with selected use cases. *Journal of Pathology Informatics*, 7(1):29, 2016.
- [11] Diederik P. Kingma and Jimmy Ba. Adam: A method for stochastic optimization, 2017.
- [12] Dongguang Li, Jacob R. Bledsoe, Yu Zeng, Wei Liu, Yiguo Hu, Ke Bi, Aibin Liang, and Shaoguang Li. A deep learning diagnostic platform for diffuse large b-cell lymphoma with high accuracy across multiple hospitals. *Nature Communications*, 11(1), November 2020.
- [13] Tsung-Yi Lin, Priya Goyal, Ross Girshick, Kaiming He, and Piotr Dollár. Focal loss for dense object detection, 2018.
- [14] Ming Y. Lu, Richard J. Chen, Jingwen Wang, Debora Dillon, and Faisal Mahmood. Semi-supervised histology classification using deep multiple instance learning and contrastive predictive coding. *CoRR*, abs/1910.10825, 2019.
- [15] Adam Paszke, Sam Gross, Francisco Massa, Adam Lerer, James Bradbury, Gregory Chanan, Trevor Killeen, Zeming Lin, Natalia Gimelshein, Luca Antiga, Alban Desmaison, Andreas Kopf, Edward Yang, Zachary DeVito, Martin Raison, Alykhan Tejani, Sasank Chilamkurthy, Benoit Steiner, Lu Fang, Junjie Bai, and Soumith Chintala. Pytorch: An imperative style, high-performance deep learning library. In H. Wallach, H. Larochelle, A. Beygelzimer, F. d'Alché-Buc, E. Fox, and R. Garnett, editors, *Advances in Neural Information Processing Systems 32*, pages 8024–8035. Curran Associates, Inc., 2019.

- [16] Mahadev Satyanarayanan, Adam Goode, Benjamin Gilbert, Jan Harkes, and Drazen Jukic. OpenSlide: A vendor-neutral software foundation for digital pathology. *Journal of Pathology Informatics*, 4(1):27, 2013.
- [17] Chetan L. Srinidhi, Seung Wook Kim, Fu-Der Chen, and Anne L. Martel. Self-supervised driven consistency training for annotation efficient histopathology image analysis. *CoRR*, abs/2102.03897, 2021.
- [18] The HDF Group. Hierarchical data format, version 5, 1997-NNNN. <https://www.hdfgroup.org/HDF5/>.
- [19] Fabiola Valvert, Oscar Silva, Elizabeth Solórzano-Ortiz, Maneka Puligandla, Marcos Mauricio Siliézar Tala, Timothy Guyon, Samuel L. Dixon, Nelly López, Francisco López, César Camilo Carías Alvarado, Robert Terbrueggen, Kristen E. Stevenson, Yasodha Natkunam, David M. Weinstock, and Edward L. Briercheck. Low-cost transcriptional diagnostic to accurately categorize lymphomas in low- and middle-income countries. *Blood Advances*, 5(10):2447–2455, May 2021.
- [20] Jianfei Zhang, Wensheng Cui, Xiaoyan Guo, Bo Wang, and Zhen Wang. Classification of digital pathological images of non-hodgkin's lymphoma subtypes based on the fusion of transfer learning and principal component analysis. *Medical Physics*, 47(9):4241–4253, July 2020.

8 Appendix

8.1 TripletNet Training Diagram from Srinidhi et. al.

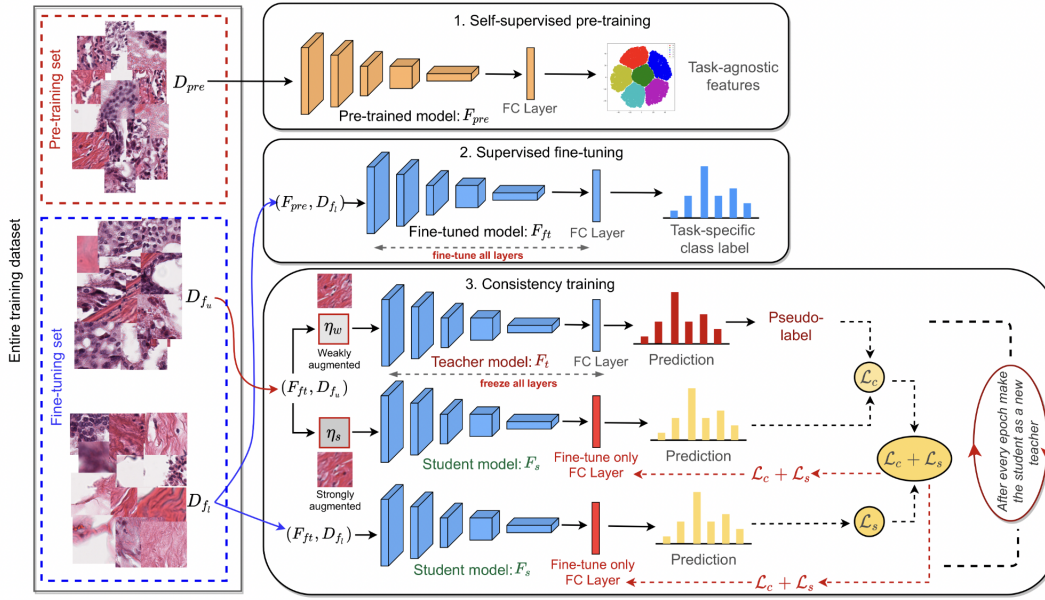


Figure 5: TripletNet Training Diagram from Srinidhi et. al.

8.2 Loss Functions

8.2.1 Weighted Crossentropy

If l_i is the number of samples (either total number of patches or number of bags in the training set corresponding to the i^{th} class), then the weights are calculated as $w_i = \frac{1/l_i}{\sum_{j=1}^N 1/l_j}$. This gives a normalized set of weights with the property that the more samples a particular label i has, the lower its weight will be. The function for weighted crossentropy is:

$$\mathcal{L}_{CE} = - \sum_{i=1}^N w_i y_i \log \hat{y}_i$$

8.2.2 Focal Loss

Focal loss was developed for object detection with the R-CNN by modifying the regular crossentropy to deal with label sparsity [13]. After initial testing with the crossentropy and the weighted crossentropy loss, we still observed that the imbalance still

adversely affected results and focal loss is a well known method beyond class weighting that helps with class imbalance. In focal loss, $-\log \hat{y}$ term is multiplied by two terms, $\alpha, (1 - \hat{y})^\gamma$, where α, γ are tunable. The function for focal loss is:

$$\mathcal{L}_{FL} = -\sum_{i=1}^N \alpha y_i (1 - \hat{y}_i)^\gamma \log \hat{y}_i$$

8.3 Confusion Matrices

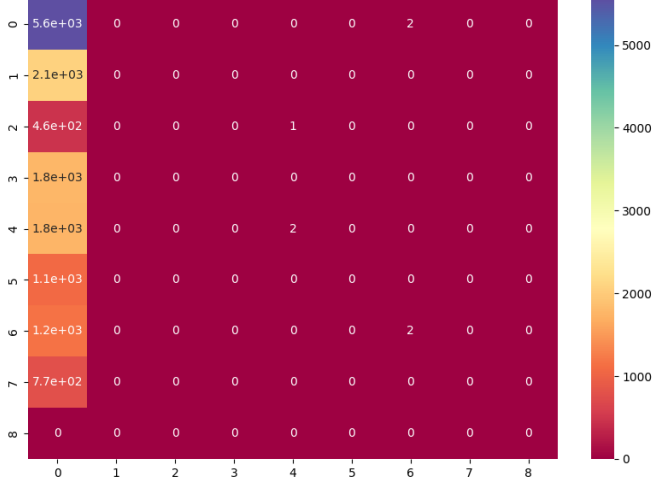


Figure 6: Confusion matrix for Naive model with focal loss (on 50 epochs). The Naive model on weighted crossentropy had the same confusion matrix.

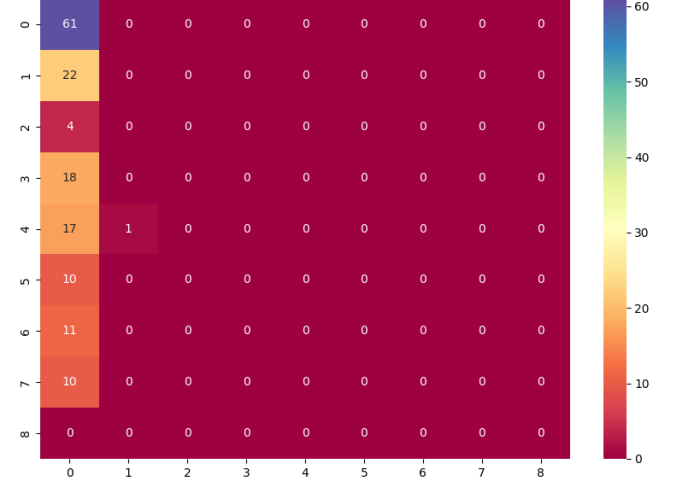


Figure 7: Confusion matrix for MIL model with focal loss (on 50 epochs)

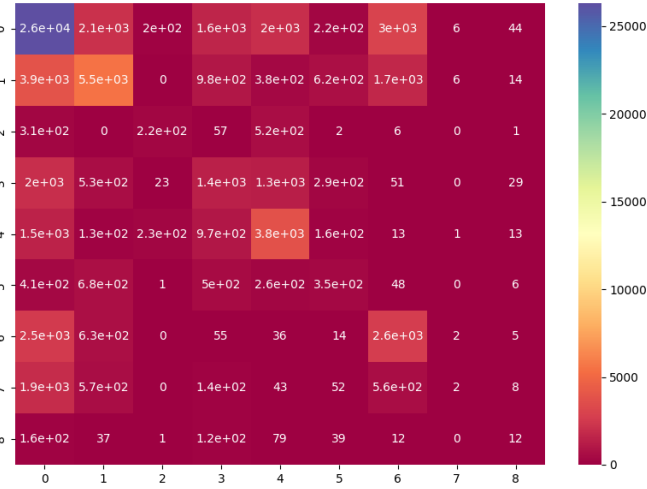


Figure 8: Confusion matrix for Naive model with end-to-end training on the training set

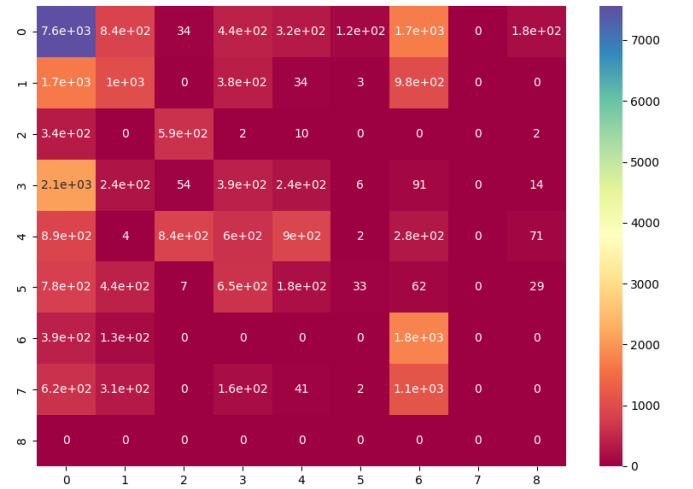


Figure 9: Confusion matrix for Naive model with end-to-end training on the validation set