# Challenge on Diagnosis of Lymphocytosis via Multiple Instance Learning

Team Name: Juju & Pedro

Pierre Clavier
ENS Paris-Saclay
pierre.clavier@ens-paris-saclay.fr

Julia Linhart
ENPC, ENS Paris-Saclay
julia.linhart@eleves.enpc.fr

## 1. Introduction

The competition aims at developing new algorithms to detect a lymphoproliferative disorder (type of cancer of the lymphocytes) in blood smears presenting lymphocytosis, by diagnosing them as either reactive or tumoral. This task takes the form of a *weakly supervised classification problem*, whose main challenges are the small sample sizes of highly unbalanced data and the only globally (i.e. bag-level) available labels.

**A Multiple Instance Learning (MIL) problem:** In addition to age, sex and lymphocyte count, a given patient is here represented by a "bag" of several microscopic subimages (or instances) of the collected blood smear. The patient will be diagnosed with cancer if at least one of the instances is tumoral. The problem is thus naturally formulated as Multiple Instance Learning : the model aims at infering labels at bag-level, based on information contained within the corresponding instances, but without having access to instance-level annotations during training (it is very difficult and time-expensive to provide such local labels).

**Difficult Data:** The problem here is that even cancerous blood smears naturally present a lot more *normal* than *tumoral* instances: a "smart" aggregation technique of those instances is thus necessary to avoid the model's ignorance towards the much less represented cancerous cases and thus optimize the learning process. Also, only very little and highly unbalanced data is available for training: 142 subjects with 44 reactive and 98 malignant cases. To avoid over-fitting, several different generalization methods (regularization, dropout out, early stopping, etc.) were used and combined (see section 2). Model performance was measured by balanced accuracy.

**Overview:** We chose to implement the *Chowder* [6] and *DeepMIL* [5] models, two different MIL approaches mentioned in [2], adapting them to our task and data. Their architecture and methodological components are described in section 2.2. We then explain the changes we made on those models to improve accuracy (section 2.3) and generalization (section 2.4). Finally, we present and discuss the choice of hyper parameters and obtained results in section 3, as well as possible improvements in section 4.

## 2. Architecture and methodological components

The baseline approach of this Kaggle Challenge is described by 3 main components: a ConvNet to extract features of the instance-level images, a linear classifier and a mean operator to aggregate predictions. We propose an adaptation and improvement of this method.

### 2.1. Feature Extraction

Adopting the same approach as in from [6], our feature extractor consists in a on ImageNet pretrained ResNet50 [3], that provides us with 2048 features per instance. Indeed, this Transfer Learning approach is particularly used nowadays to extract quality features from image data. After aggregation, a classifier can be trained to produce the desired diagnosis labels. How features and instance are aggregated, defines the different MIL approaches.

### 2.2. Chosen Models

As mentioned previously, we are dealing with highly unbalanced bags, as cancerous regions of a blood smear are naturally highly localized. A simple aggregation method such as the mean operator will thus most likely overwhelm the contribution of the disease-containing instances and greatly degrade the performance of the classifier. In order to deal with this problem, we chose to look at two different MIL approaches that use attention mechanisms to select the most relevant instances to train the classifier. Their respective architectures are presented in Figure 2. The choice of hyperparameters will be explained in section 3.1.

**Chowder:** This MIL-model uses a 1D convolution for the embedding of the 2048 extracted ResNet features. Of these sorted embedding values, only the top and bottom R entries are retained, resulting in a tensor of 2×R entries to use for diagnosis classification. This can be easily accomplished through a MinMax layer on the output of the one-dimensional convolution layer. The purpose of this method is to chose the instances that best support the presence *and the absence* of the class. A set of fully-connected layers is then placed on top of these "extreme"-instances (i.e. top instances and negative evidence) for bag-level classification. To avoid over-fitting (extremely important as the sample size is very small), l2-regularization is performed on the convolutional weights and dropout is applied before the final output layer. We also chose to use an ensemble prediction (average of multiple models that only differ by their weight-initialization) to reduce the variance of different model predictions. Indeed, different initializations lead to models with very different performances, although they all have the same architecture (see table 3).

**DeepMIL:** The in Chowder eliminated image-features (from non-extreme instances) might still provide useful information. A "soft"-elimination approach seems more appropriate: given the small sample size, we wish to use as much information as possibly available.

*DeepMIL* [5] adresses this issue with a *gated-attention system: attention-weights* for all the instances are learned and used to form a "fully-informed" bag-level prediction. Training differs from the Chowder model, as we perform early-stopping, but no regularization. To assess the epoch-choice in [2] for early-stopping, we looked at the validation (30% of the training data) loss evolution as explained in section 3.1. Using ensemble predictions is also a useful option here (see table 3).

### 2.3. Additional features

The above described models only take into account the image part of the dataset. But it also provides us with information about the age and sex of the patients, as well as the lymphocyte count of their blood smears. We chose to modify the classifiers by integrating those additional features: they are first normalized and then concatenated with the MIL-features before the multi-layer perceptron at the end of the pipeline as shown in Figures 2 and 3.

We motivate and justify this modification with the important correlation that exists between the lymphocyte counts and the positively labeled bags as shown in Figure 1. We made sure that no bias is introduced by adding the gender information (50% vs. 49% of male vs. female patients in the training data). However, our model might still over-fit on age and lymphocyte-count features.

Integrating those features to our training procedure indeed increased the training accuracy, but left us with a huge gap with respect to the test accuracy. To avoid this kind of over-fitting behavior of our models, we properly tuned the hyperparameters of the different generalization methods (l2-regularization, drop-out, ensembling and early stopping) as described in section 3.1.

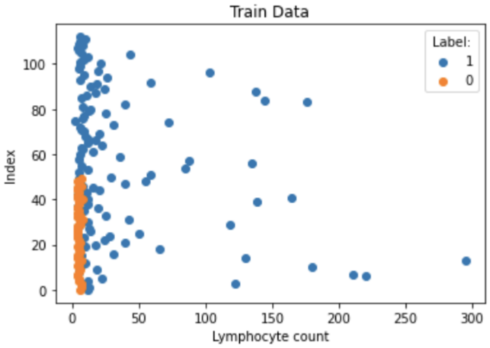


Figure 1. Correlation between the lymphocyte count and positive bag-labels. This scatter plot is obtained on the training data.

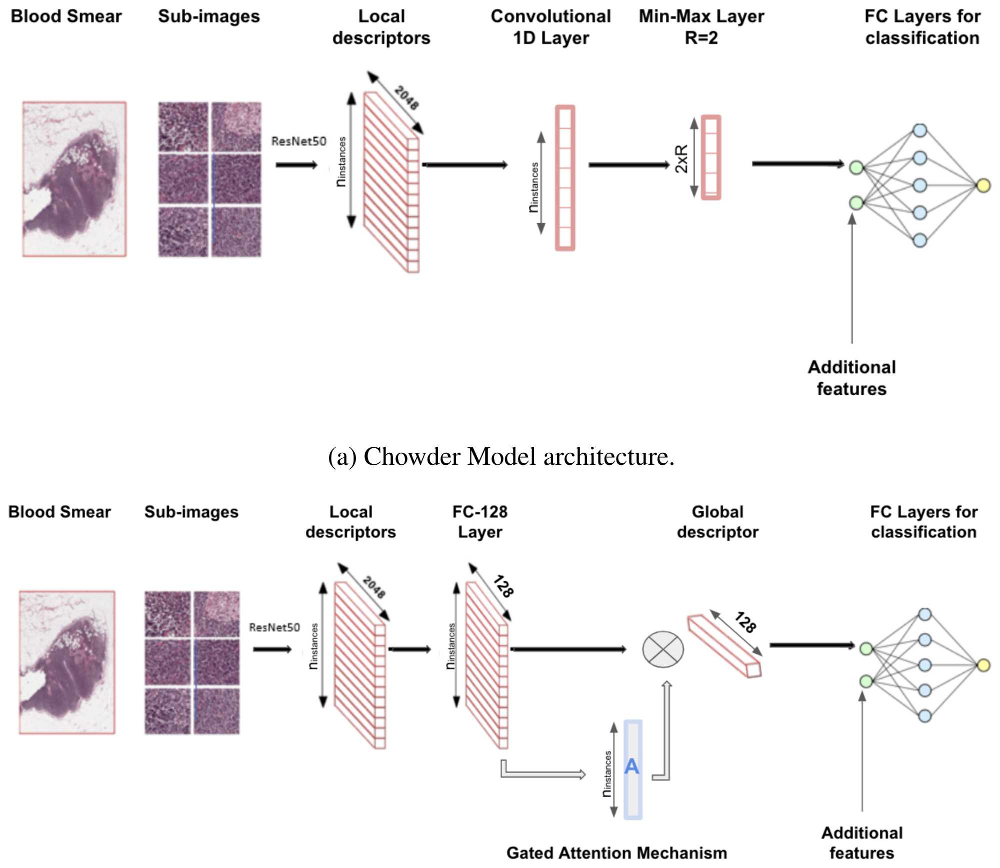
### 2.4. Variational Encoding

We also chose to use Variational Encoding in order to give a more general and flexible representation power to our model: sampling from the learned feature distribution of each instance-image, can in a certain way be seen as some kind of data augmentation. Thus, the generalization power of our model is increased, even if we have many more parameters to learn. We implemented a variational version of the DeepMIL model, whose architecture is described in Figure 3 and is trained by optimizing the original BCE-loss with an additional KL-divergence term.

Unfortunately, this method did not improve our model and test results. This could be explained by the following observation: the classifier has a direct, deterministic access to the source, as the attention weights are computed from the real features. The sampled latent variables Z however might not capture much information and the VAE's role becomes useless. The authors of [1], referred to this problem as a *bypassing phenomenon* and propose a solution using *Variational Attention*.

## 3. Model tuning and comparison

We implemented the two models described in section 2.2 using the PyTorch library. Imlementation details are shown in Tables 1 and 2. The code can be found in the attached .zip file or at `https://github.com/JuliaLinhart/DLMI_DataChallenge`. The *Readme.txt* file contains instructions to reproduce submission results available in the *results* folder. In order to correctly tune the models, 30% of the training data was used as a validation set: this enabled us to track the models' performance and prevent any significant over-fitting behavior. The chosen evaluation metric is balanced accuracy (which will be simply referred to as accuracy in the rest of the report). We also looked at the mean-loss evolution.



(a) Chowder Model architecture.

(b) DeepMIL Model architecture.

Figure 2. Description of the Chowder and DeepMIL Model architectures with integration of additional features.

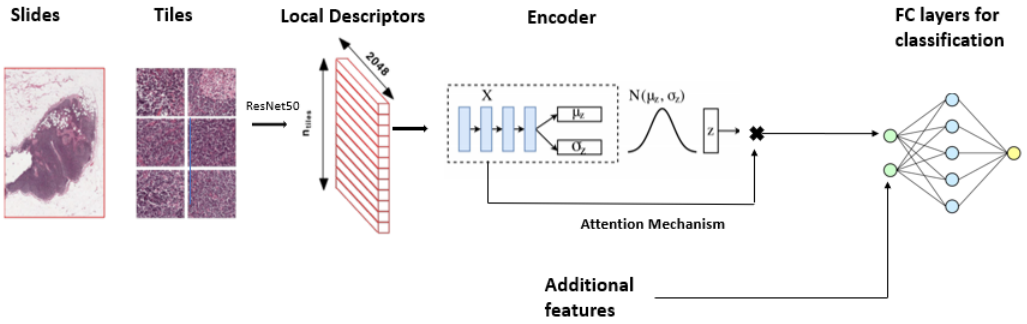


Figure 3. Description of the DeepMIL model architecture with Variational Encoding.

### 3.1. Hyperparameter selection

Both models were trained on the extracted ResNet50 features using Adam-optimizer (with default parameters) to minimize the binary cross-entropy loss over $n = 100$ epochs. A mini-batch size of 15 was chosen, as any lower value for the batch size prevented the model from correctly learning shared weights (no stable improvement of the training accuracy). We chose to use an epoch-dependent learn-

Table 1. Chowder implementation details

| Layer | Type |
|---|---|
| 1 | conv-1D |
| 2 | extreme-scores-5 + additional features |
| 3 | fc-200 + sigmoid |
| 4 | fc-100 + sigmoid |
| 5 | fc-1 + sigm |

Table 2. DeepMIL implementation details

| Layer | Type |
|---|---|
| 1 | fc-128 |
| 2 | gated-attention-128 + additional features |
| 3 | fc-128 + relu |
| 4 | fc-64 + relu |
| 3 | fc-1 + sigmoid |

ing rate initialized at $0.001$ and decreasing regularly by $40\%$ (every 20th epoch for Chowder and 10th epoch for DeepMIL) as shown in Figure 4. Indeed, a learning rate of $0.0002$ (baseline choice) led the models to get stuck in local minima, whereas a learning rate of $0.001$ prevented them from converging. Our approach enabled the models to converge to a global optimum.

To reduce variance and prevent over-fitting, we trained an ensemble of $E = 10$ networks, which only differ by their initial weights. Higher values for $E$ did not have much impact on the test results. We also applied a $p = 0.1$ dropout before the last fully connected layer of both models (any higher value prevented the model from learning because of the small sample sizes). Other generalization methods differ for both models and are described below:

**Chowder:** l2-regularization is performed on the convolutional weiths of the 1D-feature embedding. The contribution of this regularization needs to be as high as possible, without preventing the model's performance on the validation set: the value of $0.4$ was therefore chosen to give the best validation accuracy, reducing the gap to the training accuracy.

**DeepMIL:** No l2-regularization is performed: using a non-zero weight-decay led to a constant training accuracy of $0.5$. Instead, we used an early-stopping approach: looking at the evolution of the validation accuracy and mean-loss, we were able to detect the moment where continued training would lead to over-fitting (increasing training performance vs. decreasing validation performance). The number of training epochs for the DeepMIL model was therefore reduced to $n = 50$. This approach is particularly important if a constant learning rate is used and can actually be replaced by a imposing a very low learning rate value after epoch $n = 50$, preventing the model to continue the training procedure. Indeed, as mentioned previously, we chose to decrease the learning rate (initialized at $0.001$) every 10 epochs by $40\%$, resulting in a very low learning rate value (lr $< 8 \times 10^{-5}$) after $n = 50$ epochs (cf. Figure 4).

**Variational DeepMIL:** We adopted the same hyperparameter choices as for the DeepMIL model, but changed the lr-scheduler, now initilized at $0.01$ and decreasing the learning rate every $15$ epochs by $60\%$. Further experiments could be done by tuning the KL-divergence regularization of the variational loss term.

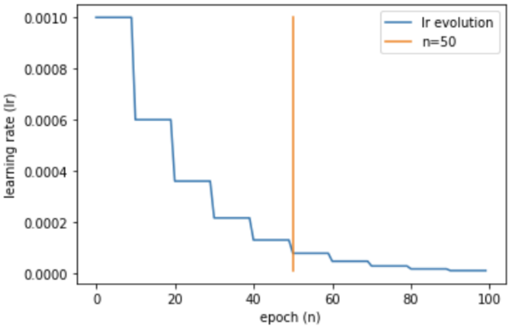


Figure 4. Evolution of the learning rate using PyTorch's Step_lr scheduler. Initial value of $0.001$, step size of $n = 10$, multiplication factor $\gamma = 0.6$.

### 3.2. Model comparison

The models were trained with optimal hyperparameters (chosen as explained in section 3.1) on 70% of the training data. The other 30% were used for evaluation purposes: we chose to only submit the results with the best validation scores. Submission results are shown in Table 3. We can see that model-ensembling slightly increased the test accuracy for both, Chowder and DeepMIL. Integrating the additional features then drastically improved their performance. Variational encoding did not improve the model's performance on the testset. This result was expected, as the model did not manage to train properly (train BA did not exceed 80%).

As mentioned in section 2.3, the tuning of generalization parameters is crucial to prevent over-fitting. Table 4 shows us the impact of l2-regularization (of the convolutional weights) on the Chowder model performance: train accuracy is reduced, but only to allow a better validation (and test) score: the gap between train and validation accuracies is largely reduced for the higher l2-reg parameter.

Our best public test score (Table 3) is obtained for a *E-DeepMIL+AF*-model, where early-stopping and dropout were crucial to obtain a satisfying (second best) validation score. We chose this model and the *E-Chowder+AF*-model for our final test scores on the private leaderboard. The latter has the best validation score. It might be an overfitting case on the validation data, but results for this model were more consistent, which should reduce the risk of overfitting on the given fraction of the test data.

|  | Chowder | DeepMIL | E-Chowder | E-DeepMIL | E-Chowder+AF | E-DeepMIL+AF | Var-DeepMIL+AF |
|---|---|---|---|---|---|---|---|
| **test BA** | 74,03% | 75,58% | 74,81% | 85,71% | 84,94% | **89,87%** | 78,96% |
| **val BA** | 72 − 82% | 83 − 88% | 80,21% | 86,81% | **91,89%** | 88,78% | 80,82% |

Table 3. **Submission Results (public leaderboard).** Balanced Accuracy (BA) for predictions on 50% of the test data. We also show the validation BA in the second row of the table. A range (over 10 different models) was given for the non-ensemble versions to emphasize the variance of the predictions, leading to a random test BA (could have been high or low). E stands for *Ensemble*, AF for *Additional Features*.

|  | l2-reg = 0.1 | l2-reg = 0.4 |
|---|---|---|
| **train BA** | 96,5% | 92,99% |
| **val BA** | 82,25% | 90,99% |
| **test BA** | 78,96% | 85,71% |

Table 4. **Impact of l2-regularization.** Results for Chowder Models with additional features, trained with different different l2-regularisation values. In both cases E=1 and the best validation BA (with corresponding train BA) over 100 epochs are presented. We can see that the regularization reduces the gap between train and validation accuracies. The test accuracy of the corresponding submission (for E=10) is also shown.

‘

## 4. Future Improvements

In this section we describe three main areas of improvement, from the simplest to the most sophisticated one.

**Local loss annotation :** The first simple idea is to take into account the local annotation loss. Indeed, negatively labeled bags contain only non-tumoral (label 0) instances. We can therefore add a local loss (i.e. at instance-level) to add this more precise information about non-tumoral instances: if some local annotations would have been provided, this apporachh would have enabled us to take them into account.

**Self Supervised Learning** Recently, self supervised learning has emerged, which is particularly adapted to medical imaging where image collection is sometimes difficult. This method is for example used in histology and similar problems to ours in [2]. These methods allow to adapt the weights of the pre-trained networks to the given context/domain and thus improves the quality (or coherence) of the extracted feature. However, those algorithms are computationally particularly expensive.

**Variational Attention**

When combined with a traditional (deterministic) attention mechanism, the variational latent space may be bypassed by the attention model, and thus becomes ineffective. This is why [1] propose a variational attention mechanism, where the attention vector is also modeled as Gaussian distributed random variables. To improve our algorithm, we can be tempted to use this algorithm because the results seem more promising with this method.

**Low rank regularization and Domain Generalization** When we have to face the problem of training on limited datasets, the DNN is lacking of generalization capability, as the trained DNN on data within a certain distribution may not be able to generalize to the data with another distribution such as a slightly different test set. To tackle this issue, domain generalization [7, 4] explores a set of source domains and often uses *Low Rank regularization* to push the network to learn interesting, shareable information over those domains. This may also be an interesting idea to explore in order to improve our performance.

## References

[1] Hareesh Bahuleyan, Lili Mou, Olga Vechtomova, and Pascal Poupart. Variational attention for sequence-to-sequence models. *arXiv preprint arXiv:1712.08207*, 2017. 2, 5

[2] Olivier Dehaene, Axel Camara, Olivier Moindrot, Axel de Lavergne, and Pierre Courtiol. Self-supervision closes the gap between weak and strong supervision in histology. *arXiv preprint arXiv:2012.03583*, 2020. 1, 2, 5

[3] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 770–778, 2016. 1

[4] Haoliang Li, YuFei Wang, Renjie Wan, Shiqi Wang, Tie-Qiang Li, and Alex C Kot. Domain generalization for medical imaging classification with linear-dependency regularization. *arXiv preprint arXiv:2009.12829*, 2020. 5

[5] Max Welling Maximilian Ilse, Jakub M. Tomczak. Attention-based deep multiple instance learning. *arXiv:1802.04712v4*, 2018. 1, 2

[6] Marc Sanselme Gilles Wainrib Pierre Courtiol, Eric W. Tramel. Classification and disease localization in histopathology using only global labels: A weakly supervised approach. *arXiv:1802.02212v2*, 2020. 1

[7] Zheng Xu, Wen Li, Li Niu, and Dong Xu. Exploiting low-rank structure from latent domains for domain generalization. In *European Conference on Computer Vision*, pages 628–643. Springer, 2014. 5