Expectation-Maximization vs Deep Sets for Histopathology Data

Anastasiia Livochka, Maryam Yalsavar, Cassandra Wong, David Evans

UNIVERSITY OF WATERLOO

EM General Idea

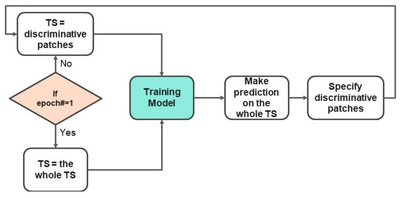
It is natural to say that patches X and their labels H were generated from:

$$P(X,H) = \prod_{i=1,N} \prod_{j=1}^{N_i} P(X_{i,j}|H_{i,j})P(H_{i,j})$$

The MLE of the unknown parameters is determined by maximizing the marginal likelihood of the observed data

$$L(\theta;X) = P(X,H|\theta)$$

Expectation Maximization Steps



M-step: we update θ to maximize the data likelihood. Assuming a uniform generative model for all non-discriminative instances we allow the simplification:

$$\theta \leftarrow argmax \prod_{x_{i,j} \in D} P(x_i, y_i|\theta)$$

E-step: estimate binary labels $H_{i,j}$ with updated model and thresholding rule. Minimum of class level and image level threshold is considered.

Importance of Second-Level Model

When aggregating patch labels to an image label, simple methods as voting and max aren't robust and do not match the decision process of pathologists.
E.g. mixed subtype of cancer might have distinct regions of other cancer subtypes.

Expt#	FL-Model	SL-Model	Test accuracy	Train Accuracy	SL-TrainData Creation	Img_size
1	Resnet18	LR	0.607	0.757	Count per class	64*64
2	Resnet18	LR	0.607	0.757	Ave-probs per class	64*64
3	Resnet18	SVM	52.38(poly) 48.81(rbf)	70.00(poly) 65.26(rbf)	Count per class	64*64
4	Resnet18	SVM	52.38(poly) 48.81(rbf)	73.68(poly) 65.26(rbf)	Ave-probs per class	64*64

Results

Weighted cross-entropy loss on train - data likelihood

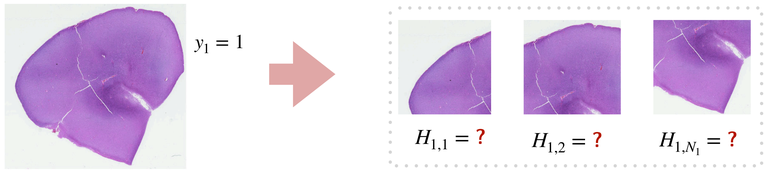
60%
attained dev accuracy

Epoch accuracies for images estimated by patch predicted labels max counts

Problem Formulation

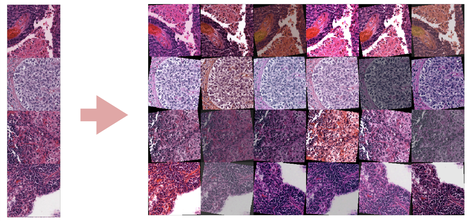
We want to classify three types of lung cancer: LUSC, MESO, LUAD. Hovewer, due to high computational costs deep learning approaches cannot be applied to Whole Slide Tissue Images. One of the most promising approaches to reduce needed resources is sampling patches from original WSI and train models based on patches.

The bag $X_1 = \{X_{1,1}, \ldots X_{1,N_1}\}$ and the set of unknown patches labels $H_1 = \{H_{1,1}, \ldots, H_{1,N_1}\}$



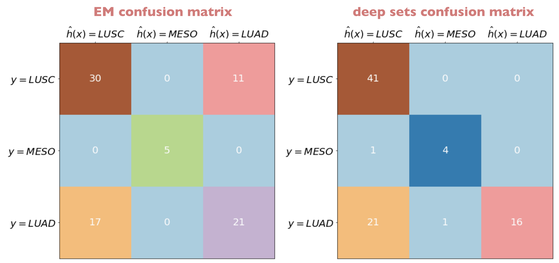
The training dataset in this study contains 72 LUAD directories, 156 LUSC directories, and 9 MESO directories. Each directory is a folder containing its patched images. We can see that the dataset is strongly unbalanced as the MESO subtype has a significantly lower number of dataset compared to others.

Patches Preprocessing



From visual inspection of data in both train an dev we noticed that sometimes images of the same class have very different brightness/contrast. Also it is natural to assume that different devices might have been used to scan lungs of different patients

Conclusions & Future Work



Both approaches turned to be very unstable during training.

Gradient optimization was frequently stack in local minima (giving the same lables to all instances) or jumping from one local minima to another.

Although on average it was easier to get better accuracy with Deep Sets.

We have many ideas that due to the lack of time are left for future work.

1. EM algorithm: experiments with percentiles for image- and class-level thresholds, experiments with network architecture and image augmentations.
2. Deep Sets: experiments with both encoder and head model architecture, training regime (learning rate, freezing layers), experiments with loss function and class weights.

Deep Sets Intuition

Because patches itself don't have a know label it is natural to approach this problem in the framework of set classification. Problems involving a set of objects have the permutation invariance property: the target value for a given set is the same regardless of the order of objects in the set.

A simple example of a permutation invariant model is

$$model(\{X_{i,1}, \ldots, X_{i,N_i}\}) = \rho(pool(\{\phi(X_{i,1}), \ldots, \phi(X_{i,N_i})\}))$$

1. Each patch $X_{i,j}$ is transformed (encoded) into some representation $\phi(X_{i,j})$
2. The representations in the set $S_i = \{\phi(X_{i,j}) : j \in J\}$ are aggregated by sum/max/avg
3. Aggregated representation is used by ρ to produce image-level prediction

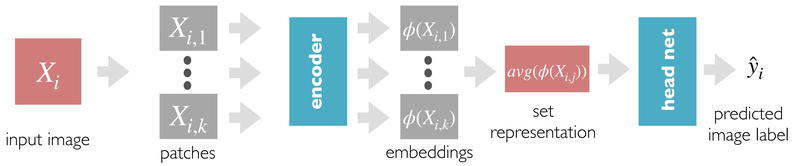
Implementation Details

Encoder: Resnet-34 pretrained on ImageNet, 512 features extracted before classifier layer

Head model: 4 fully connected layers (512, 512, 256, 64, 3) followed by BatchNorm1D and ReLU activations

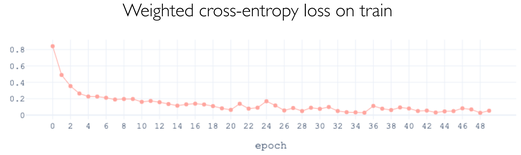
Both models are trained simultaneously with weighted cross entropy w = (0.2, 0.45, 0.35) loss and Adam optimizer with lr = 5e-5. Number of patches used for one image = 7. Max pooling for obtaining set representation.

Experiments showed that color and affine augmentation in combination with weighted loss makes training even more unstable than usual and terminates with very low accuracy (train ~ 20%).



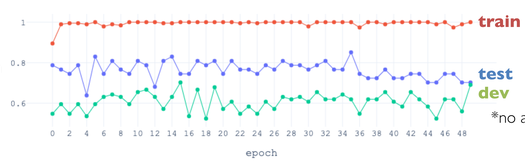
Results

Weighted cross-entropy loss on train



70%
attained dev accuracy

Epoch accuracies for images where set representation is estimated from all image patches



*no augmentation