Survival Prediction via Deep Attention-Based Multiple-Instance Learning Networks with Instance Sampling

Aliasghar Tarkhan[1], Trung Kien Nguyen[2], Noah Simon[1], Jian Dai[2]

[1]Department of Biostatistics, University of Washington, Seattle, WA, USA
[2]Imaging and Personalized Healthcare Group, Genentech, South San Francisco, CA, USA
tarkhan.arash@gmail.com, ntkien1283@gmail.com, nrsimon@uw.edu, dai.jian@gene.com

Abstract

Survival prediction via training deep neural networks with giga-pixel whole-slide images (WSIs) is challenging due to the lack of time annotation at the pixel level or patch (instance). Multiple instance learning (MIL), as a typical weakly supervised learning method, aims to resolve this challenge by using only the slide-level time. The attention-based MIL method leverages and enhances performance by weighting the instances based on their contribution to predicting the outcome. A WSI typically contains hundreds of thousands of image patches. Training a deep neural network with thousands of image patches per slide is computationally expensive and time-consuming. To tackle this issue, we propose an adaptive-learning strategy where we sample a subset of informative instances/patches more often to train the deep survival neural networks. We also present other sampling strategies and compare them with our proposed sampling strategy. Using both real-world and synthesized WSIs for survival, we show that sampling strategies significantly can significantly reduce computing time while result in no or negligible performance loss. We also discuss the benefits of each instance sampling strategy in different scenarios.

Introduction

The gold standard for diagnosing many diseases in pathology, such as prostate cancer, involves using high-resolution Whole-Slide Images (WSIs) from biopsies (Fraggetta et al. 2017; Cai et al. 2022). However, this process requires manual review by an expert, which is expensive, time-consuming, and susceptible to observer variability (Brunyé et al. 2010; Sederberg et al. 2020). Building automated systems to evaluate giga-pixel WSIs can save time and money while delivering high-quality health care (Tarkhan et al. 2021; Cui and Zhang 2021). Developing such automated tools is challenging due to the size of WSIs, as they are too large to be fed into deep neural networks directly. One immediate solution is to divide the WSI into smaller regions, usually hundreds of thousands of images of size 256×256, called patches or tiles. However, training a deep neural network with these many small images (patches) is challenging due to the lack of patch-level annotation, and it is costly and time-consuming to have a professional annotate these patches (Quellec et al. 2017; Zare et al. 2022; Farasat et al. 2022). The problem becomes even more challenging in survival prediction scenarios where the labels are continuous time-to-event, and certain patients are censored (Yao et al. 2020). It is practically impossible for a professional to assign a continuous time label to each patch.

Multiple-instance learning (MIL), a weakly supervised learning method (Dietterich, Lathrop, and Lozano-Pérez 1997; Maron and Lozano-Pérez 1998), tackles this challenge by training neural networks using only the bag-level labels. However, an upcoming challenge with the MIL problem is that not all instances (image tiles) are equally predictive of the bag label (class), and some may even relate to other classes (Liu, Wu, and Zhou 2012). Some works have considered combining the instance-level responses from a classifier to alleviate this challenge (Raffel and Ellis 2016; Ramon and Raedt 2000; Ilse, Tomczak, and Welling 2018). Among them, (Ilse, Tomczak, and Welling 2018) proposed an attention-based deep MIL framework to deal with this challenge. Their proposed framework includes (1) an attention network and (2) a classification network, both trained simultaneously. The attention network has parameters for updating the attention (importance) weights of different instances, while the classification network has parameters for the classification task. Authors in (Chen et al. 2021; Jiang, Suriawinata, and Hassanpour 2023) extended this work to survival prediction using attention networks. However, a challenge remains: They use all instances per bag across all training iterations. Since a WSI typically has hundreds of thousands of image tiles, training a neural network with all these instances is time-consuming and computationally expensive. An attention MIL network may not need to be trained with noisy or less-predictive instances. Authors in (Tarkhan et al. 2022a,b) proposed an adaptive sampling strategy for training deep MIL networks for classification tasks. They investigated different sampling strategies and showed that they reduce computational complexity.

This paper extends the concept of adaptive sampling by applying it to a more complex task: survival prediction. We propose an adaptive sampling strategy to train deep attention-based MIL survival neural networks and compare

it with other sampling strategies. Through experiments using both real-world and synthesized WSIs, we demonstrate that our sampling strategies significantly reduce computation time for survival prediction, outperforming the scenario where whole instances are used for training (Ilse, Tomczak, and Welling 2018).

MIL and Attention-Based MIL Networks

MIL Problem Formulation

Suppose there are N patients with bags of images (WSIs) $\boldsymbol{\mathcal{X}}^{(1)}$, $\boldsymbol{\mathcal{X}}^{(2)}$, ..., $\boldsymbol{\mathcal{X}}^{(N)}$ and bag-level labels $(y^{(1)}, \delta^{(1)})$, $(y^{(2)}, \delta^{(2)})$, ..., $(y^{(N)}, \delta^{(N)})$. For n-th patient, $y^{(n)} = min(t_n, c_n)$ is the observation time and is defined as time to event (t_n) or censoring (c_n) whatever happens first. $\delta^{(n)} = \mathbf{1}[t_n \leq c_n]$ is the event status which takes value 1 if the patient experienced the event (i.e., $t_n \leq c_n$), otherwise 0 (i.e., $t_n > c_n$). The bag for n-*th* patient (i.e., $\boldsymbol{\mathcal{X}}^{(n)}$) contains K_n instance images $\boldsymbol{X}_1^{(n)}$, $\boldsymbol{X}_2^{(n)}$, ..., $\boldsymbol{X}_{K_n}^{(n)}$ with unknown time labels $y_1^{(n)}$, $y_2^{(n)}$, ..., $y_{K_n}^{(n)}$. Note that we assume that the censoring is non-informative and not related to the disease complication encoded by patches $\boldsymbol{X}_k^{(n)}$, $k = 1, 2, \ldots, K_n$. Therefore, for the n-th patient, we assume that all patches have the same event status labels, i.e., $\delta_1^{(n)} = \delta_2^{(n)} = \cdots = \delta_{K_n}^{(n)} = \delta^{(n)}$. In classical supervised learning, we have $K_n = 1$, i.e., there is only one image per subject. To decrease computing time and cost, it is common to use a state-of-the-art pre-trained network such as *ResNet50* (He et al. 2015) to extract a low-dimensional embedding feature $\boldsymbol{h}_k^{(n)}$ from k^{th} instance image. After that, we have dataset $\{(\boldsymbol{h}_k^{(n)}, y^{(n)}, \delta^{(n)})$, for $n = 1, 2, \ldots N$ and $k = 1, 2, \ldots, K_n\}$. The goal of the MIL problem is to train a neural network using the bag label $(y^{(n)}, \delta^{(n)})$ and embedding features $\boldsymbol{h}_k^{(n)}$, $k = 1, 2, \ldots, K_n$. A possible approach to achieve this is to pool instances (e.g., by using element-wise maximum or average operators) to get a single aggregated feature representing each bag. Unfortunately, such pooling approaches are generally pre-calculated and not trainable (Ilse, Tomczak, and Welling 2018).

Attention-Based MIL

Attention-based MIL (Ilse, Tomczak, and Welling 2018) resolves this issue by using a combined architecture of two trainable networks: an attention network and a classification network. The attention network aggregates the embedding features as

$$\boldsymbol{h}_{bag}^{(n)} = \sum_{k=1}^{K_n} a_k^{(n)} \boldsymbol{h}_k^{(n)},$$

$$a_k^{(n)} = \frac{exp\{\boldsymbol{w}^T\big(tanh(\boldsymbol{V}\boldsymbol{h}_k^{(n)}) \odot sigm(\boldsymbol{U}\boldsymbol{h}_k^{(n)})\big)\}}{\sum_{k'=1}^{K_n} exp\{\boldsymbol{w}\big(tanh(\boldsymbol{V}\boldsymbol{h}_{k'}^{(n)}) \odot sigm(\boldsymbol{U}\boldsymbol{h}_{k'}^{(n)})\big)\}}, \tag{1}$$

where $\boldsymbol{w} \in \mathbb{R}^{L\times 1}$, $\boldsymbol{U} \in \mathbb{R}^{L\times M}$, and $\boldsymbol{V} \in \mathbb{R}^{L\times M}$ are trainable parameters included in the attention network; $tanh(.)$ and $sigm(.)$ are the element-wise hyperbolic tangent and sigmoid functions; $\odot$ is an element-wise multiplication. Note that the attentions weights are between 0 and 1 (i.e, $0 \leq a_k^{(n)} \leq 1$ and they are summed up to 1 (i.e., $\sum_{k=1}^{K_n} a_k^{(n)} = 1$). As a result, these weights can be seen as the probability vectors of a multi-nominal distribution, as we will use them in our proposed sampling strategy. Since $\boldsymbol{w}$, $\boldsymbol{U}$, and $\boldsymbol{V}$ are trainable, $a_k^{(n)}$ is trainable, and this pooling mechanism is trainable. Such a MIL pooling mechanism preserves flexibility and interpretability (see Section 2.4 in (Ilse, Tomczak, and Welling 2018)). After obtaining the bag-level aggregated feature $\boldsymbol{h}_{bag}^{(n)}$ for patient n by the attention network in (1), the survival prediction network receives it as the input and calculates a scalar value representing the bags as

$$\boldsymbol{s}_{bag}^{(n)} = \boldsymbol{W}_r^T \boldsymbol{h}_{bag}^{(n)}, \quad n = 1, 2, \ldots, N, \tag{2}$$

where $\boldsymbol{W}_r \in \mathbb{R}^{M\times 1}$ is a trainable vector. Although there are different loss functions which could be used for the sake of optimization in this paper, we choose the one called *negative log-likelihood* function (Zadeh and Schmid 2021). This loss function is based on a discrete-based survival model considering each patient's binary event status. The key quantities for survival prediction based on this model are the discrete hazard and survival functions which are calculated using by $\{(\boldsymbol{s}_{bag}^{(n)}, y^{(n)}, \delta^{(n)})$, for $n = 1, 2, \ldots N\}$. One can minimize such a loss function to update the model parameters from the attention and survival prediction networks combined. We encourage to read (Zadeh and Schmid 2021; Chen et al. 2021) for more details about the discrete survival model and how it was implemented.

In many pathology applications, there many instances within each WSI which may increase computing time and cost of training a deep neural network-based model. The following section presents different sampling strategies to overcome these possible shortcomings. The aim of these strategies is to sub-select a very limited number of instances to train the deep attention-based MIL neural networks for survival prediction.

Instance Sampling Strategies

This paper proposes an adaptive instance sampling strategy for the survival prediction using deep attention-based MIL networks and compare this strategy with the other sampling strategies. We briefly talk about them in the following sections.

Random Sampling

Training a deep neural network with whole instance per can be challenging due to memory constraint: It might not be feasible to bring all instances/images of a (batch of) patient (bag) into memory at a time. With the random sampling strategy for patient n, we randomly draw G ($G << K_n$) instances (or images) out of $\{1, 2, \ldots, K_n\}$ to train the deep neural network. This strategy has been used in the literature (Zhu, Yao, and Huang 2016; Wulczyn et al. 2020; Li et al.

2018) and demonstrated great success in reducing computing resources and time. However, different instances might have different amount of information for predicting the survival time and the random sampling strategy ignores this.

Adaptive Sampling

Not all instances have the same amount of information for predicting the survival time: Many of them might have little or no information while a few of them might be well-predictive of the survival time. Here we propose to adaptively draw G (again $G << K_n$) well-predictive instances per subject (bag) from an empirical sampling distribution (Tarkhan et al. 2022a,b). For nth patient, before training for the next epoch, we first extract the attention weights $a_k^{(n)}, k = 1, 2, \ldots, K_n$ from the forward attention network given for by (1). We then construct a multi-nominal sampling distribution with a vector of probabilities given by the attention weights as

$$\mathcal{P}^{(n)} = \left(p_1^{(n)} = a_1^{(n)},\, p_2^{(n)} = a_2^{(n)},\, \ldots,\, p_{K_n}^{(n)} = a_{K_n}^{(n)}\right). \tag{3}$$

After obtaining the sampling distribution, we propose to draw a subset of G indices from the sampling distribution $\mathcal{P}^{(n)}$ as

$$(I_1^{(n)}, I_2^{(n)}, \ldots, I_G^{(n)}) \sim \mathcal{P}^{(n)}. \tag{4}$$

With (4), instances that have higher attention weights (i.e., have higher $a_k^{(n)}$ and are more predictive of the outcome) will be chosen more often than others during training. Note that one can obtain the random sampling strategy from our proposed adaptive sampling strategy by setting $p_1^{(n)} = p_2^{(n)} = p_{K_n}^{(n)} = \frac{1}{K_n}$. After adaptively selecting the G instances for patient n, the aggregated feature is obtained by

$$\begin{aligned} \boldsymbol{h}_{bag,\,adaptive}^{(n)} &= \sum_{k \in (I_1^{(n)}, I_2^{(n)}, \ldots, I_G^{(n)})} a_{k,\,adaptive}^{(n)} \boldsymbol{h}_k^{(n)}, \\ a_{k,\,adaptive}^{(n)} &= \frac{a_k^{(n)}}{\sum_{k \in (I_1^{(n)}, I_2^{(n)}, \ldots, I_G^{(n)})} a_k^{(n)}}. \end{aligned} \tag{5}$$

The above expressions are exactly the same as (1) except need to we make sure the attention weights of the selected G instances are added to 1. After obtaining the aggregated feature based on adaptively selected G instances, we feed it into the survival prediction network and we follow the rest of procedure as before.

The estimates of the network parameters and consequently the attention weights $a_k^{(n)}$ are usually noisy (unstable) over the first couple of iterates (epochs). To minimize the effect of such instabilities on our adaptive sampling strategy, we propose to consider a few initial iterates as warm-up iterates where we use all instances to train the network. Although the estimation of sampling distribution through the forward attention network is faster than training the whole network, it may add computing overload if we do it on every epoch. To alleviate such overload, one might decide to estimate $\mathcal{P}^{(n)}$ only after every a pre-specified number of epochs, e_{update}.

Top-G Sampling

Another choice for instance sampling $top - G$ is sampling strategy, which has been used in the computational pathology literature (Campanella et al. 2019; Sharmay et al. 2021). In this sampling strategy, we choose G instances with the highest instance-level scores (here attention weights) as

$$(I_1^{(n)}, I_2^{(n)}, \ldots, I_G^{(n)}) = \underset{i_1, i_2, \ldots, i_G}{\operatorname{argmax}} \left\{ a_1^{(n)},\, a_2^{(n)},\, \ldots,\, a_{K_n}^{(n)} \right\}. \tag{6}$$

After choosing G instances, the rest of procedure will be same as what we presented for the adaptive sampling.

Dataset, Network Architectures, and Tuning Hyper-Parameters

Datasets

In this paper, we use both a real-world dataset and a synthesized dataset including WSIs to evaluate and compare different sampling strategies.

TCGA-LUSC Lung cancer is among those causing high mortality worldwide (Cai et al. 2022). The aim of The Cancer Genome Atlas Lung Squamous Cell Carcinoma Collection (TCGA-LUSC) data is to connect cancer phenotypes to genotypes by providing clinical images (Herbst, Morgensztern, and Boshoff 2018). After pre-processing and feature extraction (see (Williamson et al. 2021; Chen et al. 2021) for more details), the resulting dataset has 466 patients, with observed time until 5287 days, and a censoring rate of 58%.

Synthesized Survival Dataset Using MNIST We use the MNIST dataset (Deng 2012) to simulate multiple-instance learning survival datasets. The MNIST dataset is a standard benchmark dataset that has been used for multi-class classification purposes. It contains $n_{train} = 60,000$ and $n_{test} = 10,000$ greyscale images with size 28x28 for training and testing. These images correspond to digit numbers between 0 and 9. The reason to choose MNIST dataset is that one can achieve an accuracy of over 99% for a simple and non-MIL 10-class classification problem even with simple network architectures (Pedamonti 2018). Another reason is that we can explore a scenario for which there is a perfect signal to predict risk and survival. We create a MIL-based dataset using MNIST by following the below steps:

(a) For patient n, we select digit d_n from 0-9 with a probability of 0.1. The selected digit d_n represents the risk score for n-th patient: a higher digit value means a higher risk score which corresponds to a lower time. With the given digit (or risk value) for patient n, we generate time-to-event/censoring for individual n is proportional to its corresponding digit value d_n:

$$\begin{aligned} &y^{(n)} \sim exp(\mu = \exp(-\eta d_n)), \\ &\delta_n \sim Bernoulli(p = 1 - p_c),\ p_c = Pr(t_n > c_n). \end{aligned} \tag{7}$$

where η controls the amount of signal in our dataset: Higher η corresponds to better separation of event times for different digits. This data generation mechanism follows the assumption that the baseline hazard function follows an exponential distribution with the parameter $\lambda = 1$. We refer the readers to (Bender, Augustin, and Blettner 2005; Tarkhan and Simon 2022; Tarkhan et al. 2021) for more details on how to generate survival time using the MNIST dataset.

(b) We randomly select $10 \sim 100$ images for the selected digit,

(c) We randomly select $10 \sim 100$ partial images from all digits 0-9. Such partial images contain a maximum of 25% of the randomly cropped regions of digits,

(d) We randomly select $10000 \sim 50000$ pure noisy images whose pixels take values from the uniform distribution $\sim U(0, 1)$.

We generate the MNIST-based datasets with $N = 200$ samples. We do not pre-process the images to consider a scenario where a pre-processing mechanism does not exist. For n-th patient, we vectorize all 28×28 grey-scale images to get 784×1 feature vectors $\boldsymbol{h}_k^{(n)}$, $k = 1, 2, \ldots, K_n$.

Network Architecture

First, we consider a fully connected layer $\boldsymbol{W}_d \in \mathbb{R}^{d \times 512}$ ($d = 1024$ for TCGA-LUSC data and $d = 784$ for MNIST-based synthesized data) with the ReLU activation function to reduce the dimension feature embedding space from d to 512. For the attention network, we consider the gated attention with $\boldsymbol{U}, \boldsymbol{V} \in \mathbb{R}^{256 \times 512}$, each followed by a single

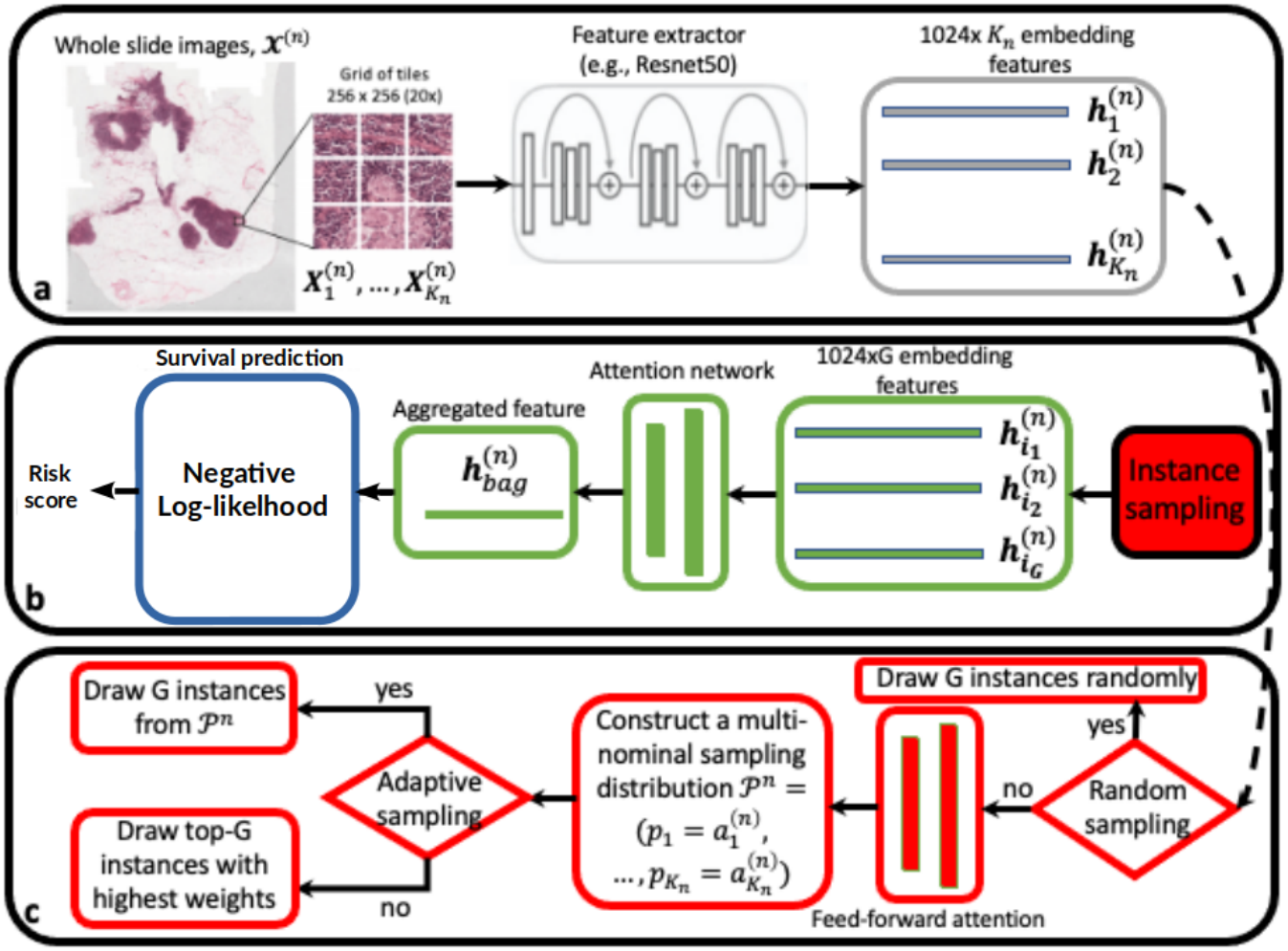

Figure 1: Different instance sampling strategies. **a: pre-processing** We sample patches from the WSI $\boldsymbol{X}^{(n)}$ and use pre-trained network (e.g., ResNet50) to extract lower-dimensional features $\boldsymbol{h}_1^{(n)}$, ..., $\boldsymbol{h}_{K_n}^{(n)}$. **b: training procedure** We use a subset of G instances (selected by strategies in panel **c**) and obtain aggregated feature $\boldsymbol{h}_{bag}^{(n)}$ using the attention network. Then, we predict the survival risk score using the survival prediction network. **c: instance sampling strategies** We use the trained (fixed) feed-forward attention network to estimate the sampling distribution $\mathcal{P}^{(n)}$ and then draw G instances out of K_n instances using different sampling strategies.

shared branch $\boldsymbol{w} \in \mathbb{R}^{256\times 1}$. For the survival prediction network, we choose a fully connected layer $\boldsymbol{W}_r \in \mathbb{R}^{512\times 1}$. We choose NLL loss function where we subdivide the observed time into 10 intervals. We use the Adam optimizer (Kingma and Ba 2017) as the optimization algorithm.

Tuning Hyper-Parameters

To find the best possible model for classification, we consider different hyper-parameters for all methods evaluated in this paper: initial learning rate with values $(10^{-4}, 10^{-3})$, regularization rate $(10^{-5}, 10^{-3})$, and dropout rate $(0.2, 0.5)$. We randomly split data into training/validation/testing datasets (80% training, 10% validation, and 10% training). We train a model on the training dataset for each combination of hyper-parameters until there is no improvement in the validation concirdance index (CI). We consider a minimum and maximum number of epochs equal to 20 and 50. We also use a stopping criterion (Prechelt 2012) with *patience*=10 epochs (after which there will be no later improvement in CI on the validation dataset) to determine when to stop training. We use the best model to maximize the validation CI and report testing CI.

Results

We compare four sampling strategies: (1) *no sampling* where we use whole instances over iterates (this is the standard attention MIL in (Ilse, Tomczak, and Welling 2018) and is used by (Williamson et al. 2021; Chen et al. 2021)), (2) *random sampling* where we randomly draw choose G instances, (3) *adaptive sampling* where we adaptively select G instances, and (4) $top-G$ *sampling* where we choose G instances with the highest attention weights. We evaluate different strategies with both TCGA-LUSC and synthesized MNIST-based survival datasets. We use the same network architecture, hyper-parameters, and tuning procedure for all methods. For all methods, we choose the minimum number of epochs as $e_{min} = 20$, the maximum number of epochs as $e_{max} = 50$, and patience $e_{patience} = 10$ for early stopping. For random and adaptive sampling methods, we initially consider 10 warm-up epochs ($e_{warm} = 10$) to train the model using whole instances. We conducted all experiments on AWS nodes with one NVIDIA Tesla T4 GPU node, 32 CPUs, and 235 GB of memory. We consider 10 repetitions of Monte Carlo simulations for splitting data into training/validation/testing, and we report *mean* $\pm$ *Standard error (SE)*. Figures 2 compares the testing concordance index and training time for TCGA-LUSC. For this dataset, all sampling strategies reduces the training time compared with no sampling strategy. The *random* and *adaptive* sampling strategies perform very close to *no sampling* and the $top-G$ *sampling* strategy perform worst. We may expect this because we used the pre-processing to exclude noisy instances or instances with little information and all remained instances have a good amount of information for predicting the survival time. The $top-G$ *sampling* strategy seems to suffer most probably due to overfitting. Figures 3 and 4 compare the testing concordance index and training time for the synthesized MNIST-based survival datasets with sample size $N = 200$ and with the signal levels of $\eta = 0.5$ and $\eta = 1.0$, respectively. We observe *adaptive* and $top-G$ sampling strategies perform very close to *no sampling* strategy and *random sampling* performs the worst. We also observe that all instance sampling strategies significantly reduce the training time. We expect this gap in training time because we assumed that there is no pre-processing step for these datasets and there are many non-informative and few informative instances. The reason *random sampling* performs worst is, with limited number of selected instances, it merely picks the informative instances. This strategy needs more epochs and time in order to to learn related patterns for the survival.

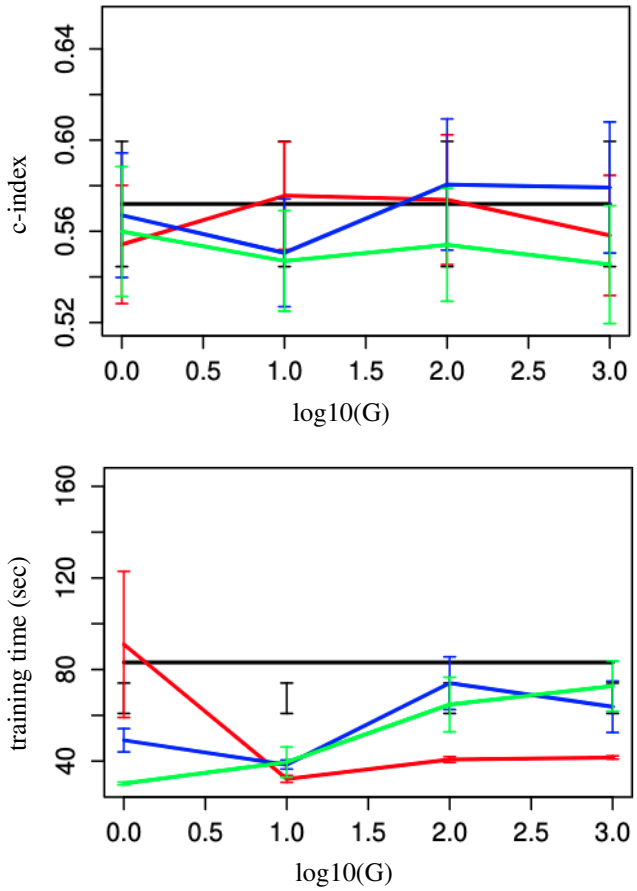

Figure 2: TCGA-LUSC; (top) average concordance index and (bottom) training time; We compare different sampling strategies: no sampling (black), random sampling (red), adaptive sampling (blue), and top-G sampling (green).

Discussion

We used both real-world (TCGA-LUSC) and synthesized (MNIST-based) datasets to investigate different instance sampling strategies for survival prediction using attention-based MIL networks. We showed that these strategies significantly reduce computing time (and hence resources) with a negligible performance loss.

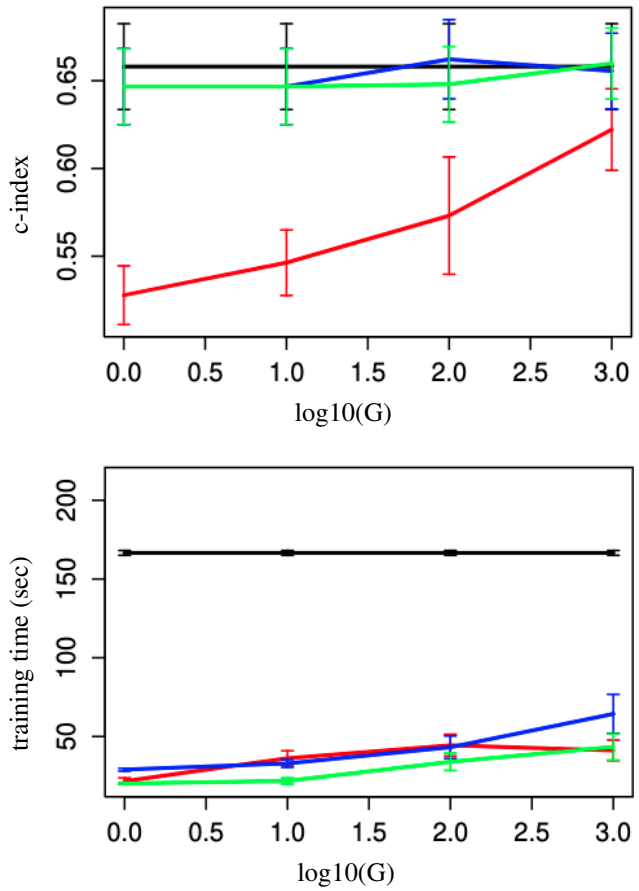

Figure 3: Synthesized MNIST-based survival data with $N = 200$ samples and $\eta = 0.5$; (top) average concordance index and (bottom) training time; We compare different sampling strategies: no sampling (black), random sampling (red), adaptive sampling (blue), and top-G sampling (green).

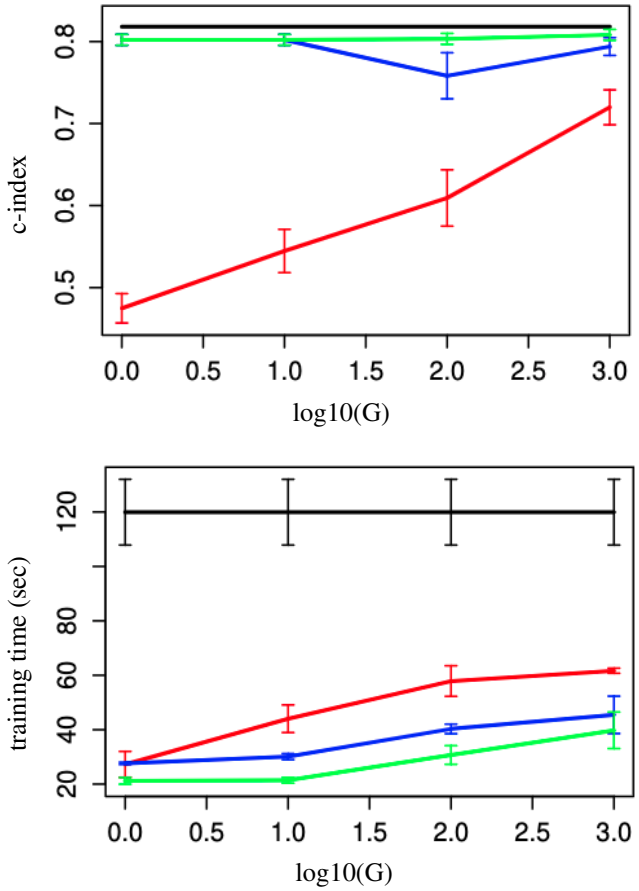

Figure 4: Synthesized MNIST-based survival data with $N = 200$ samples and $\eta = 1$; (top) average concordance index and (bottom) training time; We compare different sampling strategies: no sampling (black), random sampling (red), adaptive sampling (blue), and top-G sampling (green).

For TCGA-LUSC dataset, we used pre-processing to exclude tiles with no or little information (e.g., noise or background), and then used a pre-trained network (e.g., Resnet50) to extract low-dimensional features from the remaining image tiles. We observed that all methods perform closely because all features contain a good amount of information about the patients' survival. It is worth investigating more complicated scenarios, e.g., fine-tuning some layers of the pre-trained network to extract more distinct and predictive features out of sampled tiles. One can further examine the proposed instance sampling strategy using other real world datasets such as TCGA-LGGGBM (lower grade glioma & glioblastoma) (Network and et. al. 2008, 2015). However, we observed that this dataset lacks enough signal for survival prediction when using train/validation/test splits and return concordance index very close to 0.5.

For MNIST-based dataset, we did not use a pre-processing step and we vectorized the images to get one-dimensional images. The goal was to compare different sampling strategies without a pre-processing step. We observed that random sampling suffers significantly because only a small fraction of images contain information about the survival. We consider maximum of 50 epochs. It is worth allowing more epochs, especially for the random sampling strategy. It is also worth using more complex networks, e.g., convolutional neural networks (CNN).

Comparing training times for for different instance sampling strategies using the real-world (with pre-processing) and the synthesized (without pre-processing) datasets highlights the fact that with absence of pre-processing step, instance sampling strategies result in huge decrease in training time.

Acknowledgments

Thanks Imaging and Personalized Healthcare Group at Genentech Inc. for supporting this work.

References

Bender, R.; Augustin, T.; and Blettner, M. 2005. Generating Survival Times to Simulate Cox Proportional Hazards Models. *Statistics in medicine*, 24(11): 1713–1723.

Brunyé, T. T.; Mercan, E.; Weaver, D. L.; and Elmore, J. G. 2010. Accuracy is in the eyes of the pathologist: The visual interpretive process and diagnostic accuracy with digital whole slide images. *J. of Biomed. Info.*, 66: 171–179.

Cai, L.; Xiao, G.; Gerber, D.; Minna, J.; and Xie, Y. 2022. Lung Cancer Computational Biology and Resources. *Cold Spring Harb Perspect Med.*, 12(2): 1–23.

Campanella, G.; Hanna, M. G.; Geneslaw, L.; Miraflor, A. P.; Silva, V. W. K.; Busam, K. J.; Brogi, E.; Reuter, V. E.; Klimstra, D. S.; and Fuchs, T. J. 2019. Clinical-grade computational pathology using weakly supervised deep learning on whole slide images. *Nature Medicine*, 1–9.

Chen, R. J.; Lu, M. Y.; Weng, W.-H.; Chen, T. Y.; Williamson, D. F.; Manz, T.; Shady, M.; and Mahmood, F. 2021. Multimodal Co-Attention Transformer for Survival Prediction in Gigapixel Whole Slide Images. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 4015–4025.

Cui, M.; and Zhang, D. Y. 2021. Artificial Intelligence and Computational Pathology. *Laboratory Investigation*, 101: 412–422.

Deng, L. 2012. The MNIST Database of Handwritten Digit Images for Machine Learning Research. *IEEE Signal Processing Magazine*, 29(6): 141–142.

Dietterich, T. G.; Lathrop, R. H.; and Lozano-Pérez, T. 1997. Solving the multiple instance problem with axis-parallel rectangles. *Artificial Intelligence*, 89(1): 31–71.

Farasat, M.; Aalaei, E.; Kheirati Ronizi, S.; Bakhshi, A.; Mirhosseini, S.; Zhang, J.; Nguyen, N.-T.; and Kashaninejad, N. 2022. Signal-Based Methods in Dielectrophoresis for Cell and Particle Separation. *Biosensors*, 12(7).

Fraggetta, F.; Garozzo, S.; Zannoni, G. F.; Pantanowitz, L.; and Rossi, E. D. 2017. Routine digital pathology workflow: the catania experience. *J Pathol Inform.*, 8(51): 1–6.

He, K.; Zhang, X.; Ren, S.; and Sun, J. 2015. Deep Residual Learning for Image Recognition. arXiv:1512.03385.

Herbst, R.; Morgensztern, D.; and Boshoff, C. 2018. The biology and management of non-small cell lung cancer. *Nature*, 553: 446–454.

Ilse, M.; Tomczak, J. M.; and Welling, M. 2018. Attention-based Deep Multiple Instance Learning. arXiv:1802.04712.

Jiang, S.; Suriawinata, A. A.; and Hassanpour, S. 2023. MHAttnSurv: Multi-head attention for survival prediction using whole-slide pathology images. *Computers in Biology and Medicine*, 158: 106883.

Kingma, D. P.; and Ba, J. 2017. Adam: A Method for Stochastic Optimization. arXiv:1412.6980.

Li, R.; Yao, J.; Zhu, X.; Li, Y.; and Huang, J. 2018. Graph CNN for Survival Analysis on Whole Slide Pathological Images. In Frangi, A. F.; Schnabel, J. A.; Davatzikos, C.; Alberola-López, C.; and Fichtinger, G., eds., *Medical Image Computing and Computer Assisted Intervention – MICCAI 2018*, 174–182. Cham: Springer International Publishing. ISBN 978-3-030-00934-2.

Liu, G.; Wu, J.; and Zhou, Z.-H. 2012. Key instance detection in multi-instance learning. In *Proc. of the Asian Conference on Machine Learning*, volume 25 of *Proc. of Machine Learning Research*, 253–268. PMLR.

Maron, O.; and Lozano-Pérez, T. 1998. A Framework for Multiple-Instance Learning. In Jordan, M.; Kearns, M.; and Solla, S., eds., *Advances in Neural Information Processing Systems*, volume 10. MIT Press.

Network, C. G. A. R.; and et. al. 2008. Comprehensive Genomic Characterization Defines Human Glioblastoma Genes and Core Pathways. *Nature*, 455: 1061—1068.

Network, C. G. A. R.; and et. al. 2015. Comprehensive, Integrative Genomic Analysis of Diffuse Lower-Grade Gliomas. *N. Engl. J. Med.*, 372: 2481—2498.

Pedamonti, D. 2018. Comparison of non-linear activation functions for deep neural networks on MNIST classification task. *arXiv:1804.02763 [cs.LG]*, 1–5.

Prechelt, L. 2012. *Early Stopping — But When?*, 53–67. Berlin, Heidelberg: Springer Berlin Heidelberg. ISBN 978-3-642-35289-8.

Quellec, G.; Cazuguel, G.; Cochener, B.; and Lamard, M. 2017. Multiple-instance learning for medical image and video analysis. *IEEE Reviews in Biomedical Engineering*, 10: 213–234.

Raffel, C.; and Ellis, D. P. W. 2016. Feed-Forward Networks with Attention Can Solve Some Long-Term Memory Problems. arXiv:1512.08756.

Ramon, J.; and Raedt, L. D. 2000. Multi instance neural networks. ICML Workshop on Attribute-value and Relational Learning, 53–60.

Sederberg, M.; Liem, B.; Tarkhan, A.; Gessel, T.; LaCourse, M.; and Latzka, E. 2020. Brief Ultrasound-Aided Teaching to Improve the Accuracy and Confidence of Resident Musculoskeletal Palpation. *Phys Med Rehabil*, 12(4): 391–396.

Sharmay, Y.; Ehsany, L.; Syed, S.; and Brown, D. E. 2021. HistoTransfer: Understanding Transfer Learning for Histopathology. In *2021 IEEE EMBS International Conference on Biomedical and Health Informatics (BHI)*, 1–4.

Tarkhan, A.; Nguyen, T. K.; Simon, N.; Bengtsson, T.; Ocampo, P.; and Dai, J. 2022a. Attention-Based Deep Multiple Instance Learning with Adaptive Instance Sampling. In *2022 IEEE 19th International Symposium on Biomedical Imaging (ISBI)*, 1–5.

Tarkhan, A.; Nguyen, T. K.; Simon, N.; and Dai, J. 2022b. Investigation of Training Multiple Instance Learning Networks with Instance Sampling. In Xu, X.; Li, X.; Mahapatra, D.; Cheng, L.; Petitjean, C.; and Fu, H., eds., *Resource-Efficient Medical Image Analysis*, 95–104. Cham: Springer Nature Switzerland. ISBN 978-3-031-16876-5.

Tarkhan, A.; and Simon, N. 2022. An online framework for survival analysis: reframing Cox proportional hazards model for large data sets and neural networks. *Biostatistics*. Kxac039.

Tarkhan, A.; Simon, N.; Bengtsson, T.; Nguyen, K.; and Dai, J. 2021. Survival Prediction Using Deep Learning. In *Proc. of AAAI Spring Symposium on Survival Prediction - Algorithms, Challenges, and Applications 2021*, volume 146 of *Proc. of Machine Learning Research*, 207–214. PMLR.

Williamson, M. Y. L. D. F. K.; Chen, T. Y.; Chen, R. J.; Barbieri, M.; and Mahmood, F. 2021. Data-efficient and weakly supervised computational pathology on whole-slide images. *Nat Biomed Eng*, 5: 555—-570.

Wulczyn, E.; Steiner, D. F.; Xu, Z.; Sadhwani, A.; Wang, H.; Flament-Auvigne, I.; Mermel, C. H.; Chen, P.-H. C.; Liu, Y.; and Stumpe, M. C. 2020. Deep learning-based survival prediction for multiple cancer types using histopathology images. *PLOS ONE*, 15(6): e0233678.

Yao, J.; Zhu, X.; Jonnagaddala, J.; Hawkins, N.; and Huang, J. 2020. Whole slide images based cancer survival prediction using attention guided deep multiple instance learning networks. *Medical Image Analysis*, 65: 101789.

Zadeh, S. G.; and Schmid, M. 2021. Bias in Cross-Entropy-Based Training of Deep Survival Networks. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 43(9): 3126–3137.

Zare, F.; Aalaei, E.; Zare, F.; Faramarzi, M.; and Kamali, R. 2022. Targeted drug delivery to the inferior meatus cavity of the nasal airway using a nasal spray device with angled tip. *Computer Methods and Programs in Biomedicine*, 221: 106864.

Zhu, X.; Yao, J.; and Huang, J. 2016. Deep convolutional neural network for survival analysis with pathological images. In *2016 IEEE International Conference on Bioinformatics and Biomedicine (BIBM)*, 544–547.