Learning Not to Learn: Training Deep Neural Networks with Biased Data Supplementary Document

Byungju Kim[1] Hyunwoo Kim[2] Kyungsu Kim[3] Sungjin Kim[3] Junmo Kim[1]
School of Electric Engineering, KAIST, South Korea[1]
Beijing Institute of Technology[2], Samsung Research[3]
{byungju.kim,junmo.kim}@kaist.ac.kr
{hwkim}@bit.edu.cn, {ks0326.kim,sj9373.kim}@samsung.com

Formulation of $\mathcal{L}_{MI}(\theta_f, \theta_h)$

Here, we re-formulate regularization loss, $\mathcal{L}_{MI}(\theta_f, \theta_h)$, with more rigorous notations. If it is not specifically mentioned, the numbering of equations indicates that of supplementary material (this document). Our regularization loss is based on the mutual information between feature embedding and bias. It can be expressed as follows:

$$\mathcal{I}(b(X); f(X)) = H(b(X)) - H(b(X)|f(X)). \tag{1}$$

The Eq. (1) is the identical equation with Eq. (4) in the paper. The optimization problem to minimize the mutual information can be written as

$$\begin{aligned}\arg\min_{\theta_f} \mathcal{I}(b(X); f(X)) &= \arg\min_{\theta_f} H(b(X)) - H(b(X)|f(X)) \\ &= \arg\min_{\theta_f} -H(b(X)|f(X)) \\ &= \arg\min_{\theta_f} \mathbb{E}_{\tilde{F}\sim P_{f(X)}(\cdot)}[-H(b(X)|f(X) = \tilde{F})] \\ &= \arg\min_{\theta_f} \mathbb{E}_{\tilde{x}\sim P_X(\cdot)}[-H(b(X)|f(X) = f(\tilde{x}))].\end{aligned} \tag{2}$$

By sampling $b(X)$, the conditional entropy in Eq. (2) is reformulated as

$$\begin{aligned}-H(b(X)|f(X) = f(\tilde{x})) &= \mathbb{E}_{\tilde{b}\sim P_{b(X)|f(X)}(\cdot|f(\tilde{x}))}[\log P_{b(X)|f(X)}(\tilde{b}|f(\tilde{x}))] \\ &= \mathbb{E}_{\tilde{b}\sim Q_{b(X)|f(X)}(\cdot|f(\tilde{x}))}[\log Q_{b(X)|f(X)}(\tilde{b}|f(\tilde{x}))] \\ &\quad s.t. \quad Q_{b(X)|f(X)}(b|f) = P_{b(X)|f(X)}(b|f) \text{ for all } b \text{ and } f.\end{aligned} \tag{3}$$

By substituting the Eq. (3) for Eq. (2), we have

$$\begin{aligned}\arg\min_{\theta_f} \mathcal{I}(b(X); f(X)) &= \arg\min_{\theta_f} \mathbb{E}_{\tilde{x}\sim P_X(\cdot)}[\mathbb{E}_{\tilde{b}\sim Q_{b(X)|f(X)}(\cdot|f(\tilde{x}))}[\log Q_{b(X)|f(X)}(\tilde{b}|f(\tilde{x}))]] \\ &\quad s.t. \quad Q_{b(X)|f(X)}(b|f) = P_{b(X)|f(X)}(b|f) \text{ for all } b \text{ and } f.\end{aligned} \tag{4}$$

This is the identical with Eq. (5) in the main paper. Then, the $\mathcal{L}_{MI}$ can be reformulated as follows:

$$\begin{aligned}\mathcal{L}_{MI} = \; &\mathbb{E}_{\tilde{x}\sim P_X(\cdot)}[\mathbb{E}_{\tilde{b}\sim Q_{b(X)|f(X)}(\cdot|f(\tilde{x}))}[\log Q_{b(X)|f(X)}(\tilde{b}|f(\tilde{x}))]] \\ &+ \mu D_{KL}(P_{b(X)|f(X)}||Q_{b(X)|f(X)}).\end{aligned} \tag{5}$$

Now, we parametrize the $Q_{b(X)|f(X)}(b|f)$ as $h(b|f;\theta_h)$, and try to make $h(b|f;\theta_h)$ close to $P_{b(X)|f(X)}(b|f)$ by supervised learning. Therefore, the $\mathcal{L}_{MI}(\theta_f, \theta_h)$ is

$$\begin{aligned}\mathcal{L}_{MI}(\theta_f, \theta_h) = \ & \mathbb{E}_{\tilde{x}\sim P_X(\cdot)}[\mathbb{E}_{\tilde{b}\sim h(\cdot|f(\tilde{x}))}[\log h(\tilde{b}|f(\tilde{x}))] \\ & + \mu\mathcal{L}_c(b(\tilde{x}), h(f(\tilde{x})))],\end{aligned} \tag{6}$$

where $b(\tilde{x})$ is a one-hot label vector for bias in image $\tilde{x}$.

Cleaning Procedure for IMDB Face Dataset

To clean the noisy label from the IMDB Face dataset, we used networks pretrained with Adience dataset. We estimated the age and gender for all the individuals in the IMDB dataset. We discarded an image of the estimations that do not match with the ground truth labels. Unlike IMDB dataset, the age label of Adience dataset is provided in terms of interval: (0-2), (4-6), (8-13), (15-20), (25-32), (38-43), (48-53), and (60-). To determine whether the age estimation matches the ground truth label or not, we set a maring M. With estimated age (L, H) for an image, we consider the age estimation matches for age label of the image if the estimation is within range of (L-M, H+M).

Color Bias in MNIST Dataset

To intentionally plant color bias to MNIST dataset, every image is colored. Ten colors were selected and each color is assigned to a digit category. The colors and corresponding categories are enumerated in Table 1.

Digit	Color Name	Mean Color	Sampled Mean - Train	Sampled Mean - Test
0	Crimson	(220, 20, 60)	(214, 39, 76)	(148,134,116)
1	Teal	(0,128,128)	(29,127,127)	(151,136,116)
2	Lemon	(253,233, 16)	(225,211, 40)	(147,127,116)
3	Bondi Blue	(0,149,182)	(29,129,194)	(152,133,115)
4	Carrot orange	(237,145, 33)	(221,128, 54)	(147,132,115)
5	Strong Violet	(145, 30,188)	(143, 43,184)	(152,134,112)
6	Cyan	(70,240,240)	(72,219,219)	(148,129,118)
7	Your pink	(250,197,187)	(223,186,186)	(148,134,120)
8	Lime	(210,245, 60)	(201,221, 63)	(151,133,115)
9	Maroon	(128, 0, 0)	(127, 28, 28)	(145,133,116)

Table 1. Mean colors for sampling and sampled mean colors. Although the colors are presented with integers in [0, 255], they were normalized into [0, 1]. The sampled mean colors of test images show that the test set is independent of the color bias unlike the training set

Originally, 20 distinct colors are proposed.[1] We have selected ten colors with minor modification. With the mean colors, a color for each image is sampled following Algorithm 1.

Algorithm 1 Sample a color for each image.

```
Require: mean color $(r^m, g^m, b^m) \in \mathbb{R}^3$, variance $\sigma^2$
  Initialize sampled color $\mathcal{C}$ = []
  for $c^m$ in $[r^m, g^m, b^m]$ do
    while True do
      Sample a color $c \sim \mathcal{N}(c^m, \sigma^2)$
      if $0 < c < 1$ then
        Append $c$ to $\mathcal{C}$
        break
      end if
    end while
  end for
```

[1] https://sashat.me/2017/01/11/list-of-20-simple-distinct-colors/

Training images were colored depending on their digit labels. For example, if an image to colorize is from category zero, the color is sampled with mean color (220, 20, 60). Therefore, images from the same digit category are colorized with similar colors. In contrast, colors of the test images should be independent of their categories. To this end, a mean color is sampled uniformly in advance to the algorithm 1. As a result of sampling, the mean colors of each category are presented in Table 1.