Appendix of
Advancing Brain Imaging Analysis Step-by-step via Progressive Self-paced Learning

Anonymous

Anonymous

1 Proof

In PSPD, the paced curriculum is divided into two folds: based on the current and the past state of the model. For each term, it can be obtained as follows:

$$L = \frac{1}{N} \sum_{i}^{N} w_i l_i + R_\lambda(w_i) \tag{1}$$

Through the concurrent optimization of the model parameters denoted as θ and the importance weights represented by w utilizing an Alternative Optimization Strategy (AOS) with a gradually increasing age parameter, the capability to automatically integrate additional samples is augmented. Moreover, given fixed model parameters θ the optimal problem becomes

$$\min_{w_i \in [0,1]} w_i l_i + R_\lambda(w_i) \tag{2}$$

The optimal solutions for the hard and soft strategies are provided by the following propositions.

Proposition 1. The definitions of $R_\lambda^{hard}(w_i, l_i)$ and the closed-form solutions $w_i^{hard}(l_i, \lambda)$ are given by

$$R_\lambda^{hard}(w_i, l_i) = -\lambda w_i; w_i^{hard}(l_i, \lambda) = \begin{cases} 1, & \text{if } l_i < \lambda \\ 0, & \text{if } l_i \geq \lambda \end{cases} \tag{3}$$

Proof. For the hard regularizer $R_\lambda^{hard}(w_i, l_i)$, the optimization problem becomes

$$\min_{w_i \in [0,1]} w_i l_i - \lambda w_i = (l_i - \lambda) w_i \tag{4}$$

If $l_i < \lambda$, the minimum is achieved when $w_i = 1$. Conversely, if $l_i \geq \lambda$, the minimum is achieved when $w_i = 0$. Therefore, the optimization result precisely corresponds to the threshold defined in Equation 3.

Proposition 2.The definitions of $R_\lambda^{soft}(w_i, l_i)$ and the closed-form solutions $w_i^{soft}(l_i, \lambda)$ are given by

$$R_\lambda^{soft}(w_i, l_i) = \lambda(\frac{1}{2} w_i^2 - w_i); w_i^{soft}(l_i, \lambda) = \begin{cases} 1 - \frac{1}{\lambda} w_i, & \text{if } l_i < \lambda \\ 0, & \text{if } l_i \geq \lambda \end{cases} \tag{5}$$

Proof. For the soft regularizer $R_\lambda^{soft}(w_i, l_i)$, the optimization problem becomes

$$\min_{w_i \in [0,1]} w_i l_i - \lambda(w_i - \frac{1}{2} w_i^2) = \frac{\lambda}{2} w_i^2 - (\lambda - l_i) w_i \tag{6}$$

If $l_i \geq \lambda$, the minimum is achieved when $w_i = 0$. However, if $l_i < \lambda$, the optimum is determined by differentiating the function with respect to w_i and setting the result to zero, giving

$$w_i = 1 - \frac{1}{\lambda} l_i \tag{7}$$

The optimization result is precisely the threshold obtained by Equ.5

2 Pseudo-Code

Algorithm 1: Progressive Self-paced Learning

Input: Training Dataset $D_{train} = \{(x_i, y_i), ...\}$, Student model S_θ, Teacher model T_θ, λ_φ-scheduler $\lambda_\varphi(t)$, λ_w-scheduler $\lambda_w(t)$

Output: Parameters θ of student model

for $Epoch = 1, ..., T$ **do**
  Update λ_φ and λ_w by $\lambda_\varphi \leftarrow \lambda_\varphi(t)$ and $\lambda_w \leftarrow \lambda_w(t)$;
  Determine the current curriculum by S_θ on dataset D using $l_i^w = L_{CE}(p^S(x_i), y_i)$;
  Compute the current curriculum weight $w \leftarrow w(l_i^w, \lambda)$;
  if $Epoch = 1$ **then**
    Train the student model S_θ by the L_{PCL} with weight w
  else
    Determine the past curriculum by S_θ on dataset D using $l_i^\varphi = L_{CE}(p^T(x_i), y_i)$;
    Compute the past curriculum weight $\varphi \leftarrow \varphi(l^\varphi, \lambda)$;
    Train the student model S_θ by L_{PCL} and L_{PCD} with the weights w, φ ;
    Update the teacher model $T_\theta \leftarrow S_\theta$;
  end
end
return parameters θ of student model S_θ