Note: Semi-Supervised Semantic Segmentation Using Unreliable Pseudo-Labels

Sinkoo

April 10, 2022

1 Introduction

Unreliable pixels remain unused because they get confused to those classes with high confidence, but authors thinks they are useful to deal with those classes with low confidence by taking them as negative samples.
This paper proposed a framework U^2PL, authors employ the per-pixel entropy to separate reliable and unreliable data, reliable samples are used to give pseudo labels and unreliable predictions are pushed into a memory bank. And employing a queue for each category to balance the the number of negative samples of each category. They also adjust the threshold as model becoming more accurate.

2 Method

2.1 Overview

U^2PL has 2 models of the same architecture: teacher and student. The student's weight is updated with common practice and teachers is through EMA updated with student's weights. Each model consists of a CNN based encoder h, a decoder with a segmentation head f, and a representation head g
At each training epoch, equally sample B labeled samples B_l and B unlabeled samples B_l.
Total loss: $L = L_s + \lambda_u L_u + \lambda_c L_c$
where L_s and L_u represent supervised loss and unsupervised loss applied on labeled images and unlabeled images respectively, and L_c is the contrastive loss to make full use of unreliable pseudo-labels.

$$\mathcal{L}_s = \frac{1}{|\mathcal{B}_l|} \sum_{(\mathbf{x}_i^l, \mathbf{y}_i^l) \in \mathcal{B}_l} \ell_{ce}(f \circ h(\mathbf{x}_i^l; \theta), \mathbf{y}_i^l), \quad (2)$$

$$\mathcal{L}_u = \frac{1}{|\mathcal{B}_u|} \sum_{\mathbf{x}_i^u \in \mathcal{B}_u} \ell_{ce}(f \circ h(\mathbf{x}_i^u; \theta), \hat{\mathbf{y}}_i^u), \quad (3)$$

$$\mathcal{L}_c = -\frac{1}{C \times M} \sum_{c=0}^{C-1} \sum_{i=1}^{M} \log \left[\frac{e^{\langle \mathbf{z}_{ci}, \mathbf{z}_{ci}^{+} \rangle / \tau}}{e^{\langle \mathbf{z}_{ci}, \mathbf{z}_{ci}^{+} \rangle / \tau} + \sum_{j=1}^{N} e^{\langle \mathbf{z}_{ci}, \mathbf{z}_{cij}^{-} \rangle / \tau}} \right],$$

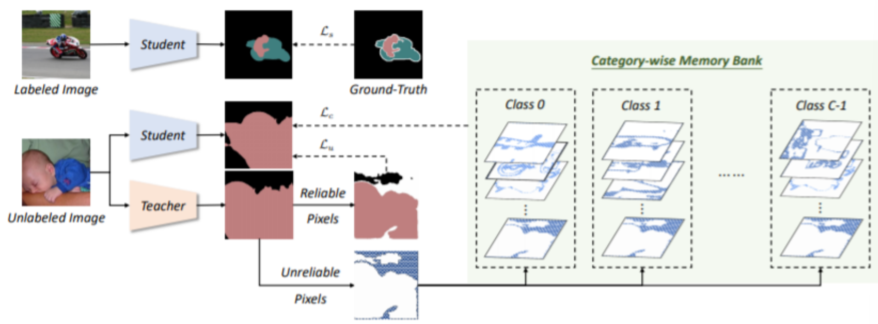

Figure 3. **An overview of our proposed U²PL method.** U²PL contains a student network and a teacher network, where the teacher is momentum-updated with the student. Labeled data is directly fed into the student network for supervised training. Given an unlabeled image, we first use the teacher model to make a prediction, and then separate the pixels into reliable ones and unreliable ones based on their entropy. Such a process is formulated as Eq. (6). The reliable predictions are directly used as the pseudo-labels to advise the student, while each unreliable prediction is pushed into a category-wise memory bank. Pixels in each memory bank are regarded as the negative samples to the corresponding class, which is formulated as Eq. (4).

3 Pseudo-Labeling

The entropy is computed by:

$$H(p_{i,j}) = -\sum_{c=0}^{C-1} p_{i,j}(c) log p_{i,j}(c)$$

where C is the number of classes, $p_{i,j} \in R^C$ as the softmax probabilities generated by the segmentation head of the teacher model for the $i-th$ unlabeled image at pixel j. Then the pseudo label is :

$$\hat{y}_{ij}^{u} = \begin{cases} \arg\max_{c} p_{ij}(c), & \text{if } \mathcal{H}(\mathbf{p}_{ij}) < \gamma_t, \\ \text{ignore}, & \text{otherwise,} \end{cases}$$

Dynamic Partition Adjustment Adjust unreliable pixels' proportion α_t with linear strategy every epoch:

$$\alpha_t = \alpha_0 (1 - \frac{t}{totalepoch})$$

Then set threshold γ_t as the quantile corresponding to α_t.

Adaptive Weight Adjustment The weight λ_u for the loss is defined as the reciprocal of the percentage of pixels with entropy smaller than threshold γ_t in the current mini-batch multiplied by a base weight η:

$$\lambda_u = \eta \cdot \frac{|\mathcal{B}_u| \times H \times W}{\sum_{i=1}^{|\mathcal{B}_u|} \sum_{j=1}^{H \times W} \mathbb{1}\left[\hat{y}_{ij}^u \neq \text{ignore}\right]},$$

4 Using Unreliable Pseudo-Labels

Anchor Pixels $A_c^l = \{z_{i,j} | y_{i,j} = c, p_{i,j}(c) > \delta_p\}$
δ_p denotes the positive threshold. z is the representation. $A_c^u = \{z_{i,j} | \hat{y}_{i,j} = c, p_{i,j}(c) > \delta_p\}$
Then for class c the anchors is : $A = A_l + A_u$.

Positive Samples Center of all possible anchors:

$$z_c^+ = \frac{1}{|A_c|} \sum_{z_c \in A_c} z_c$$

Negative Samples For labeled image: $n_{i,j}^l = \mathbf{1}[y_{i,j} \neq c] * \mathbf{1}[0 \leq O_{i,j}(c) < r_l]$ where $O_{i,j} = argsort(p_{i,j})$. Obviously, we have $O_{i,j}(argmaxp_{i,j}) = 0$ and $O_{i,j}(argminp_{i,j}) = C - 1$. r_l is the low rank threshold and is set to 3.
For unlabeled image: $n_{i,j}^u = \mathbf{1}[H(p_{i,j}) > \gamma_t] * \mathbf{1}[r_l \leq O_{i,j}(c) < r_h]$
where r_h is the high rank threshold and is set to 20.
Finally, the set of negative samples of class c is : $N = \{z_{i,j} | n_{i,j}(c) = 1\}$

5 Category-wise Memory Bank

Store the negative samples for class c.

Algorithm 1: Using Unreliable Pseudo-Labels

Initialize $\mathcal{L} \leftarrow 0$;
Sample labeled images $\mathcal{B}_l$ and unlabeled images $\mathcal{B}_u$;
for $\mathbf{x}_i \in \mathcal{B}_l \cup \mathcal{B}_u$ **do**
  Get probabilities: $\mathbf{p}_i \leftarrow f \circ h(\mathbf{x}_i; \theta_t)$;
  Get representations: $\mathbf{z}_i \leftarrow g \circ h(\mathbf{x}_i; \theta_s)$;
  for $c \leftarrow 0$ **to** $C - 1$ **do**
    Get anchors $\mathcal{A}_c$ based on Eq. (11);
    Sample M anchors: $\mathcal{B}_A \leftarrow \texttt{sample}\ (\mathcal{A}_c)$;
    Get negatives $\mathcal{N}_c$ based on Eq. (16);
    Push $\mathcal{N}_c$ into memory bank $\mathcal{Q}_c$;
    Pop oldest ones out of $\mathcal{Q}_c$ if necessary;
    Sample N negatives: $\mathcal{B}_N \leftarrow \texttt{sample}\ (\mathcal{Q}_c)$;
    Get $\mathbf{z}^+$ based on Eq. (12);
    $\mathcal{L} \leftarrow \mathcal{L} + \ell(\mathcal{B}_A, \mathcal{B}_N, \mathbf{z}^+)$ based on Eq. (4);
  end
end
Output: contrastive loss $\mathcal{L}_c \leftarrow \frac{1}{|\mathcal{B}| \times C} \mathcal{L}$

6 Summary

This paper propose a semi-supervised semantic segmentation framework U2PL by including unreliable pseudo-labels into training. But the training if this method in time consuming. And the most valuable idea of this paper I think is the importance of unreliable pixels may not useful in the probable classed but useful in the less probable classes.