MixMatch-UDA-ReMixMatch-FixMatch

Introduction

1. Published timeline: MixMatch -> UDA -> ReMixMatch -> FixMatch

2. Main techniques:

- Consistency regularization
- Entropy minimization

Consistency Regularization

- Where and how to input noise?

  - Via Data Augmentation
  - Input noise at middle of training
  - Add noise at input pictures in the beginning
  - Modify the network structure (**Dropout**)

- How to measure consistency?

  - KL Divergence
  - L2 Loss(more strict constraint according to the author of temporal ensembling)

Entropy Minimization

When temperature=0, *temperature sharpening=Pseudo label*

Pseudo label has simple implementation because it does not have *'temperature'* as hyperparameter.

Methods of entropy minimization:

1. Temperature sharpening

- MixMatch
- UDA
- ReMixMatch

2. Pseudo label

- FixMatch

Compare four of the SOTA Methods

For **entropy minimization** task:

- MixMatch：平均 K 次 weak augmentation（如 shifting 和 flipping）的 predictions，然后经过 temperature sharpening；
- UDA：一次 weak augmentation 的 prediction，然后经过 temperature sharpening；
- ReMixMatch：一次 weak augmentation 的 prediction，然后经过 distribution alignment，最后经过 temperature sharpening；
- FixMatch：一次 weak augmentation 的 prediction，然后 argmax 得到 hard label（pseudo label）。

For **consistency regularization** task:

MixMatch、UDA、ReMixMatch 和 FixMatch 都利用 data augmentation 改变输入样本来注入 noise，不同的是 data augmentation 的具体方式和强度：

- MixMatch：一次 weak augmentation 得到 prediction，这就和正常的监督训练一样，只是 unlabeled loss 用的是 L2 而已；
- UDA：一次 strong augmentation（RandAugment） 得到 prediction；
- ReMixMatch：多次 strong augmentation（CTAugment）得到 predictions，然后同时参与 unlabeled loss 的计算，即一个 unlabeled instance 一个 step 多次增强后计算多次 loss；
- FixMatch：一次 strong augmentation（RandAugment 或 CTAugment）得到 prediction。

Comparing **loss** for unlabeled data:

- MixMatch：L2 loss；
- UDA：KL divergency；
- ReMixMatch：cross entropy（包括自监督的 rotation loss 和没有使用 mixup 的 pre-mixup unlabeled loss）；
- FixMatch：带阈值的 cross entropy。

Methods to prevent overfitting in MixMatch and UDA:

1. MixMatch uses **mixup**
2. UDA uses **Training Signal Annealing(TSA)**

Training Signal Annealing(TSA) [2]

TSA prevents overfitting by **removing labeled samples that the model is already confident** about from the training set.

Specifically, examples are removed from the training set when the model's predicted probability of the correct class **exceeds a certain threshold**.

The threshold can be changed dynamically from $\frac{1}{K}$ to 1. This means only **towards the end** of training is the model allowed to **see every example** in the dataset.

This means that the model is **gradually allowed to become more confident** and see more examples = receive more training signal as training progresses, hence the term "Training Signal Annealing".

FixMatch-The Winner

FixMatch simplify MixMatch, UDA and ReMixMatch and achieves better performance:

1. temperature sharpening -> pseudo label
2. cross entropy with threshold:
   在计算 unlabeled loss 时，对 prediction 的 confidence 超过阈值的 unlabeled instance 才算入 unlabeled loss，这样使得 unlabeled loss 的权重可以固定

Reference

[1]半监督学习MixMatch、UDA、ReMixMatch、FixMatch
[2]mlexplained