SaaS: Speed as a Supervisor for Semi-supervised Learning

Safa Cicek, Alhussein Fawzi, Stefano Soatto

**Semi-supervised learning (SSL)**

- In SSL, one is given some labeled and unlabeled data.
- The goal is to train a classifier, in hopes of it performing better if trained on the labeled data alone.

**Motivation**

- The key idea of our approach is to use speed of convergence as an inference criterion for the value of the unknown labels for SSL.
- Supervision quality correlates to learning speed.

**Cumulative loss**

- To quantify learning speed, we use the cumulative loss in a fixed time (epoch) interval:
  \[
  E_t = \frac{1}{T} \sum_{t=1}^{T} \log f_{\theta_t}(x_t^{(t)}) \bigg| y_{\theta_t}(x_t^{(t)})
  \]

**Cumulative loss as a criterion for posterior**

- Cumulative loss can be written as a function of unknown labels posterior to be used as a criterion.
  \[
  \mathcal{L}(P_t) = \frac{1}{T} \sum_{t=1}^{T} \sum_{x_{\theta_t}(x_t^{(t)})} \left[ \log f_{\theta_t}(x_t^{(t)}) \bigg| y_{\theta_t}(x_t^{(t)}) \right]
  \]

**Overall optimization with entropy**

- Minimizing the entropy of label estimates on unlabeled data is common in SSL literature.
  \[
  H_Q(w) = \frac{1}{N} \sum_{i=1}^{N} - \sum_{P_i} \log f_{\theta_{P_i}}(x_i^{(i)})
  \]

**Algorithm**

- In the beginning of each outer epoch, label estimates are projected to the probability simplex; the posterior initialized randomly.
- The inner loop performs a few epochs of SGD to measure learning speed (cumulative loss) while keeping the label posterior fixed.
- The outer loop then applies a gradient step to update the unknown-label posterior.
- After each update, the weights are reset.
- After the label posterior converges, we select the maximum a-posteriori estimate and proceed with training as if fully supervised in the second phase.

**Can we just minimize cumulative loss and get correct labels?**

- Supervision quality correlates with learning speed in expectation not in every realization.
- Avoiding degenerate solutions
- Weights trained with unknown labels should have almost zero training loss on (augmented) labeled data.
- Posterior of label estimates should live in probability simplex.
- Cumulative loss should be small for augmented unlabeled data.

**Entropy as an additional regularizer**

- This optimization problem has many trivial, degenerate solutions (Zhang et al., 2016). In SaaS, label posterior minimizing the cumulative loss is found. Weights of the network are not learnable in expectation.

**Weights are not learned in the first phase of SaaS.**

- Error rates achieved by SaaS.
- Comparison of SaaS to other state-of-the-art SSL algorithms.
- Left: SaaS achieves better generalization with more unlabeled data.
- Middle: SaaS finds labels training on which is faster.
- Right: By using smaller batch size, SaaS achieves better generalization with the cost of low GPU utilization and slow training. A trick we use to improve the computational cost is to use Langevin dynamics with larger batches.

**Results**

- Error rates achieved by SaaS.
- Comparison of SaaS to other state-of-the-art SSL algorithms.

**References**