Deep Classifier Mimicry without Data Access

November 7, 2023 9:08 PM

Note: the paper is related to knowledge distillation only (no CL aspect). It uses pretrained models in terms of Teacher models

★ **Main idea:**

What is knowledge distillation (KD)?
It's a technique to transfer the knowledge from a (typically larger, more complex) teacher model to a (typically smaller, simpler) student model by *training the student to mimic the teacher's predictions, feature responses, or other inferrable quantities from the learned function.*

Example of KD:

$$\mathcal{L}(\boldsymbol{x}_i, y_i) = \lambda_1 \underbrace{\mathrm{CE}(y_i, p(\boldsymbol{z}_i^{\mathrm{S}}, 1))}_{\text{match true labels}} + \lambda_2 \underbrace{\mathrm{CE}(p(\boldsymbol{z}_i^{\mathrm{T}}, \tau), p(\boldsymbol{z}_i^{\mathrm{S}}, \tau))}_{\text{match teacher soft labels}},$$

The first term in the loss function encourages the student to predict the ground truth labels, while the second term (KD part) tries to match the softened output of the teacher.

According to the paper, most existing KD methods **might** 1) require access to the original training data (as the student needs to be trained on them), which is not practical, **or** 2) sometimes they use generative models to generate synthetic data that approximates the original data's distribution but this is also not feasible most of the time, **or** 3) require matching architectures between the teacher and student.

★ **Overall idea:**
The paper introduces a solution called "Contrastive Abductive Knowledge Extraction" (CAKE). CAKE is a method that can be applied to various machine learning models (i.e., it's a model-agnostic), and it doesn't require access to the original training data.

'Abductive knowledge extraction' refers to *'distilling' the decision boundary of the teacher only*. That is, not all the knowledge needs to be distilled from the teacher, only the decision boundary that we need the student to mimic.

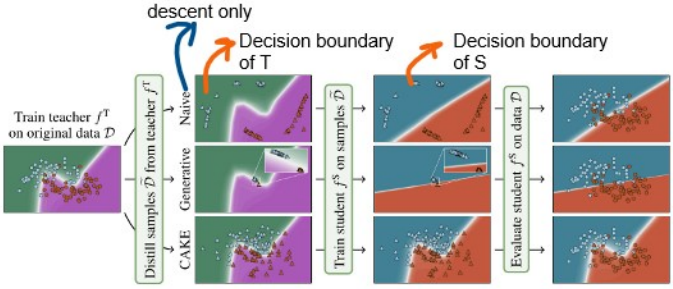

Figure 1: Comparison of naive, generative, and CAKE methods for knowledge distillation on the two-moons dataset. The background visualizes teacher (green/purple) and student (blue/red) decision functions, juxtaposed with original data (○) and synthesized samples (△). Naive and generative methods often converge to similar local minima, inducing an ineffective student decision function. In contrast, CAKE generates samples across the entire decision-relevant region, resulting in a student model that accurately learns the data decision function if trained exclusively on its synthetic samples.

CAKE works by generating synthetic data pairs through a process known as contrastive diffusion. These synthetic data pairs are directed towards opposite sides of a teacher model's decision boundary. In other words, they are designed to help the student model understand how to make decisions in a way that is similar to the original teacher model, without needing to see the original data.

The paper mentions the concept of "symbiosis:
"Contrastive pull" ensures that the student model learns from samples that closely resemble the teacher's decision-making process. To achieve this, CAKE generates pairs of noisy synthetic samples and move them closer to the decision boundary. This is done intuitively by considering two samples from different classes (or sets in multi-class scenarios) and pulling them toward each other until their predicted labels are swapped. This process ensures that the synthetic samples are guided to a region along the decision boundary.

$$\mathcal{L}_{\mathrm{contr}}(\boldsymbol{x}_i, \boldsymbol{x}_j) = \mathbb{1}\left[y_i \neq y_j\right] \left\| f^{\mathrm{T}}(\boldsymbol{x}_i) - f^{\mathrm{T}}(\boldsymbol{x}_j) \right\|_2^2$$

This loss function encourages synthetic samples from different classes to be pulled toward each other if their predicted labels differ, effectively promoting their alignment along the decision boundary.

"Induced noise" scatters samples across the decision boundary, helping the student model to explore and learn from relevant areas along the boundary. To address this, CAKE introduces noise during the sample update.

- The contrastive term already pushes samples closer to the boundary, and the noise introduced effectively disperses them in parallel to the decision boundary.
- The noise is introduced through stochastic optimization methods such as Stochastic Gradient Descent (SGD) and common step size schedules.
- This noise helps optimize the synthetic samples by dispersing them along the decision boundary, ensuring they don't collapse into a single region.

CAKE vs LAKE:
In addition to CAKE, the paper proposes another version which is called LAKE.
LAKE (Langevin Abductive Knowledge Extraction):

- LAKE is introduced as a more principled formulation for introducing noise into the synthesis procedure.
- It incorporates noise through Langevin dynamics based diffusion, generating samples from noisy gradients of the input.
- Langevin dynamics is a diffusion process that will converge samples according to the true distribution defined by the loss landscape, especially as both the number of iterations T goes to infinity and the step size $\eta(t)$ goes to zero.

$$\boldsymbol{x}_i^{t+1} = \boldsymbol{x}_i^t + \underbrace{\eta(t)\ \nabla_{\boldsymbol{x}} \mathcal{L}(\boldsymbol{x}_i^t)}_{\text{gradient update}} + \underbrace{\sqrt{2\,\eta(t)}\ \boldsymbol{\varepsilon}_i^t}_{\text{noise term}} \quad \text{with} \quad \boldsymbol{\varepsilon}_i^t \sim \mathcal{N}(0, \mathbf{I})$$

(step size)

- This diffusion process disperses the samples along the decision boundary to prevent them from collapsing.
- While theoretically the process works as $T \rightarrow \infty$ and $\eta(t) \rightarrow 0$," the presence of explicit Gaussian noise in the diffusion process may not be necessary according to some recent findings.

CAKE/LAKE adds this term to the loss, too.

$$\mathcal{L}_{\mathrm{TV}}(\boldsymbol{x}) = \sum_{i=1}^{H} \sum_{j=1}^{W} \|\boldsymbol{x}_{i,j} - \boldsymbol{x}_{i-1,j}\| + \|\boldsymbol{x}_{i,j} - \boldsymbol{x}_{i,j-1}\|$$

In addition to the strict premise of not having access to the original training data, the text acknowledges that there is often existing information about the data domain. Even in the absence of direct access to real data, the purpose and domain of application for a pre-trained model are typically evident. This auxiliary knowledge can be integrated into the sample synthesis process through the use of data priors. Intuitively, this prior mirrors our expectation that inputs are images, and we thus expect depicted concepts to be locally consistent

and we thus expect depicted concepts to be locally consistent.

★ **Total loss objective:**

They compute the extraction loss "L" as a weighted mixture of L_KD, L_contr, and L_TV. Full algorithm is shown to the right.

Algorithm 1 Contrastive Abductive Knowledge Extraction

Require: teacher f^{T}, iterations T, #mini-batches M of N samples, schedule η, priors $p(\boldsymbol{x}), p(y)$

procedure CAKE($f^{\mathsf{T}}, T, M, N, \eta, p(\boldsymbol{x}), p(y)$)
  for $m = 1$ to M **do** ▷ Number of mini-batches
    Initialize $\widetilde{\mathcal{D}}_m^{t=0} \leftarrow \left\{\left(\widetilde{\boldsymbol{x}}_1^{t=0}, \widetilde{y}_1\right), \ldots, \left(\widetilde{\boldsymbol{x}}_N^{t=0}, \widetilde{y}_N\right)\right\}$, where $\widetilde{\boldsymbol{x}}_i^{t=0} \sim p(\boldsymbol{x})$ and $\widetilde{y}_i \sim p(y)$
    for $i = 1$ to N **do** ▷ Number of synthetic samples per mini-batch
      for $t = 1$ to T **do** ▷ Number of iterations
        $\boldsymbol{z}^{\mathsf{T}} \leftarrow f^{\mathsf{T}}\left(\widetilde{\boldsymbol{x}}_i^t\right)$ ▷ Forward pass through teacher
        $l \leftarrow \mathcal{L}\left(\widetilde{\boldsymbol{x}}_i^t, \boldsymbol{z}^{\mathsf{T}}, \widetilde{y}_i, \widetilde{\mathcal{D}}_m^t\right)$ ▷ Compute extraction loss
        $\widetilde{\boldsymbol{x}}_i^{t+1} \leftarrow \widetilde{\boldsymbol{x}}_i^t - \eta(m)\, \nabla_{\boldsymbol{x}} l$ ▷ Update synthetic samples
  return $\widetilde{\mathcal{D}} = \bigcup_{m=1}^{M} \widetilde{\mathcal{D}}_m^T$

★ **Experiments and results:**
The paper has many great experiments that I recommend going through. I just wanted to show examples of the synthetic images generated by CAKE RN teacher.

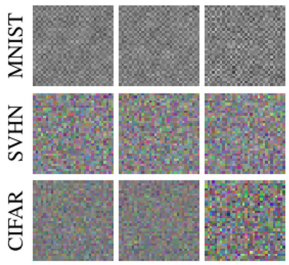

Figure 4: Synthetic samples generated from a ResNet teacher by CAKE on various datasets, demonstrating no visual resemblance with original training data.

CAKE's main premise of extracting abductive knowledge also entails that the data distribution is not closely mimicked. This implies that generated synthetic samples do not resemble original data. In fact, as depicted in Fig. 4 for three datasets, samples look rather noisy. Intuitively, they seem to look more like commonly found adversarial attacks (noise).