AdalInf: Adaptive Inference for Resource-Constrained Foundation Models

Zhuoyan Xu, Khoi Duc Nguyen, Preeti Mukherjee, Somali Chaterji, Yingyu Liang, Yin Li

Motivation

Foundation Model

Source “On the opportunities and risks of foundation models.” (2021)

Take-Home Message

We propose AdalInf—an adaptive inference framework that dynamically allocates and executes different parts of foundation models to reduce computation costs.

Key Intuition

- Existing large pretrained models have built-in redundancy, since modern training techniques adopt aggressive regularization (e.g., stochastic depth). Such redundancy allows us to treat a model as a collection of execution branches.
- Different execution branches can be tailored for runtime conditions, thereby achieving adaptive inference.

Contribution

- Our framework AdalInf learns a scheduler to decide on the branch to execute, based on a compute budget as well as the input data.
- We conduct preliminary experiments on CIFAR and ImageNet using pre-trained ResNet and CLIP models. We show AdalInf can achieve varying accuracy and latency trade-offs in response to the input data and the latency budget, outperforming baselines.

Experiments

Experimental Setup:

- Model:
  - ResNet18, ResNet32 pretrained on CIFAR100
  - CLIP (ViT-B) pretrained on LAION-400m
- Dataset:
  - CIFAR100
  - ImageNet

Results on ResNet pretrained on CIFAR100.

- Baseline: Look-up-table baseline.
- Upper Curve: The upper curve of the baseline.
- Full FT: Results on fully finetune the ResNet.
- LoRA: LoRA finetune on ResNet.
- BlockDrop: results in [ZTA+18]

Results on ViT encoder of CLIP pretrained on LAION-400m.

- Baseline: Look-up-table baseline constructed in look-up-table
- Upper Curve: The upper curve of the baseline.

Problem Formulation

- Foundation model: \( f_{\theta}(x) \)
- light-weighted scheduler: \( g_{\beta}(\cdot) \)
- Given latency requirement \( \bar{M} \), have execution plan
- Prediction \( \hat{f}(x, p) \), actual latency \( \bar{M}(x, p) \)
- Loss:
  \[
  \mathcal{L} = \mathcal{L}_{\text{CE}}(y, \hat{f}(x, p)) + \lambda \mathcal{L}_{\text{MACS}}(\bar{M}, M)
  \]
  \[
  \mathcal{L}_{\text{MACS}}(\bar{M}, M) = \max\{0, \bar{M}(x, p) - M\}
  \]