# related work

Title:MODEL COMPRESSION WITH GENERATIVE ADVER

SARIAL NETWORKS

source:斯坦福 ICLR2019

## Contributions

1. We propose GAN-assisted model compression (GAN-MC), a simple approach to improving teacher-student compression by augmenting the compression set with GAN data.
2. On CIFAR-10 image classifification, we show GAN-MC consistently improves student test accuracy for a variety of deep neural network teacher-student pairings and two popular compression objectives.
3. For random forest teachers, we demonstrate 25 to 336-fold reductions in execution and storage costs with less than 1.2% loss in test performance across a suite of real-world tabular datasets.
4. We introduce a new Compression Score for evaluating the quality of GAN-generated datasets and illustrate its advantages over the popular Inception Score on CIFAR-10.(2016,xception score)

## Related Work

**Ba & Caruana (2014)**在**hinton**之前，提出用**l2**损失函数， **hinton**提出了新的损失函数

**Method**

用**AC-GAN**做分类器 **Odena et al. (2017).**

**source:2022，cvf**加州大学 **Yoshitomo Matsubara**

contributions

- We propose a new training objective for feature compression in split computing that allows us to use a learned entropy model for bottleneck quantization in conjunction with knowledge distillation.
- Our approach significantly outperforms seven strong baselines from the split computing and (neural) image compression literature in terms of rate-distortion performance (with distortion measuring a supervised error) and in terms of end-to-end latency.
- Moreover, we show that a single encoder network can serve multiple supervised tasks, including classification, object detection, and semantic segmentation.

## Dynamic Convolution: Attention over Convolution Kernels

source（2020,微软）

问题：现有的压缩技术中，限制条件越多，即使最先进的CNN(mobilenet v3)也无法满足要求

(This paper)提出了一种新的算子设计，称为动态卷积，以增加可忽略的额外 FLOP 的表示能力

动态卷积使用一组 K 个并行卷积核 {~Wk,~bk} 而不是每层使用单个卷积核

Related work中关于本工作和Dynamic Deep Neural Networks区别

1.其他工作使用动态的网络结构，静态的卷积核，本文用动态的卷积核，静态的网络架构

2.本文不需要额外去控制，注意力机制嵌入在每一层

**Methods(DY-CNNs)**: 目标：在有效的神经网络范围内，更好的权衡网络性能和计算压力

### 3.1 Preliminary: Dynamic Perceptron

公式推导见笔记本

### Section Ⅳ two insignts of training deep dy-cnns

一**.**约束注意力输出**Πk(x)**可以促进注意力模型的学习

**Edge2Train: A Framework to Train Machine Learning Models（SVMs) on Resource-Constrained IoT Edge Devices**

**(IoT '20: Proceedings of the 10th International Conference on the Internet of Things，06 October 2020)**

在这项工作中，提供了 Edge2Train，该框架使资源稀缺的边缘设备能够在本地和离线重新训练 ML 模型

This work：在MCU上离线retrain

Contributions：

• We provide the functions for Edge2Train, which are realized through C++ implementations of our algorithms. Using these functions, users can train models (SVMs) offline on MCUs using live data from their IoT use cases. These functions also enable on-board inference and model evaluations.

• The implementation blocks of our Edge2Train fuse with the device's IoT application to continuously improve analytic results by training using the evolving real-world data.

## A PROGRAMMABLE APPROACH TO NEURAL NETWORK COMPRESSION

不需要手动调整压缩的参数

Contribution:

1.It presents CONDENSA, a new framework for programmable neural network compression. CONDENSA

supports the expression of the overall compression strategy in Python using operators provided by its compression library. Since each strategy is a Python function, users are able to programmatically compose elementary schemes to build much more complex and practically interesting schemes.

2. It presents a novel sample-effificient algorithm based on Bayesian optimization (B.O.) in CONDENSA for automatically inferring optimal sparsities based on a user-provided objective function. Given CONDENSA's ability to support the expression of meaningful highlevel objective functions—for example, the throughput (images/sec) of a convolutional neural network—users are freed from the burden of having to specify compression hyperparameters manually.

3. It demonstrates the effectiveness of CONDENSA on three image classifification and language modeling tasks, resulting in memory footprint reductions of up to 188× and runtime throughput improvements of up to 2.59× using at most 10 samples per search.