TinyML and Efficient Deep Learning Computing

Song Han
Massachusetts Institute of Technology
Model Compression & TinyML

Cloud AI  →  Mobile AI  →  AIoT

- **Cloud AI**
  - Memory: 32GB
  - Computation: TFLOPs

- **Mobile AI**
  - Memory: 4GB
  - Computation: G-TFLOPs

- **AIoT**
  - Memory: 100KB
  - Computation: MFLOPs
Efficient Deep Learning

- less computation
- fewer engineers
- less data
Training a single AI model can emit as much carbon as five cars in their lifetimes

Deep learning has a terrible carbon footprint.

by Karen Hao  June 6, 2019

The artificial-intelligence industry is often compared to the oil industry: once mined and refined, data, like oil, can be a highly lucrative commodity. Now it seems the metaphor may extend even further. Like its fossil-fuel counterpart, the process of deep learning has an outsize environmental impact.

Common carbon footprint benchmarks

in lbs of CO2 equivalent  Strubell et al.

- Roundtrip flight b/w NY and SF (1 passenger)  1,984
- Human life (avg. 1 year)  11,023
- American life (avg. 1 year)  36,156
- US car including fuel (avg. 1 lifetime)  126,000
- Transformer (213M parameters) w/ neural architecture search  626,155
- Evolved Transformer  626,155

Transformer — widely used in machine translation

- "Nice to meet you"
- "Encantada de conocerte"
- "很高兴见到你"
- "Freut mich, dich kennenzulernen"

Efficient NLP on mobile devices enable real time conversation between speakers using different languages.

Transformer — widely used in machine translation

- "Nice to meet you"
- "Encantada de conocerte"
- "很高兴见到你"
- "Freut mich, dich kennenzulernen"
Once-for-All Network: Decouple Model Training and Architecture Design

once-for-all network
Once-for-All Network: Decouple Model Training and Architecture Design

Once-for-all network
Once-for-All Network: Decouple Model Training and Architecture Design

Once-for-all network

Once-for-all, ICLR’20
Once-for-All Network: Decouple Model Training and Architecture Design

Once-for-all network

...
Progressive Shrinking

- Elastic Resolution
- Elastic Kernel Size
- Elastic Depth
- Elastic Width

Full Partial

Once-for-all, ICLR’20
Progressive Shrinking

- Elastic Resolution
- Elastic Kernel Size
- Elastic Depth
- Elastic Width

Full → Full → Full → Full
Partial → Partial → Partial → Partial

Once-for-all, ICLR’20
Train Once, Get Many

Train Once, Get Many

Train Four Times, Get Four

Top-1 ImageNet Acc (%)

OFA

MobileNetV3

70.0

67.4

70.4

73.3

76.1

75.2

67

71

73

75

77

6

9

12

15

18

21

24

Samsung Note10 Latency (ms)

Once-for-all, ICLR’20
Accelerate 3D Deep Learning & LiDAR Processing

MinkowskiNet: 3.4 FPS

SPVNAS (Ours): 9.1 FPS

Our accuracy ranks 1st on the SemanticKitti leaderboard

<table>
<thead>
<tr>
<th>Approach</th>
<th>Paper</th>
<th>Code</th>
<th>mIoU</th>
<th>Classes (IoU)</th>
</tr>
</thead>
<tbody>
<tr>
<td>SPVNAS</td>
<td>🍀</td>
<td>67.0</td>
<td>✔</td>
<td>✔</td>
</tr>
<tr>
<td>TORNADOnet</td>
<td>🍀</td>
<td>63.1</td>
<td>✔</td>
<td>✔</td>
</tr>
<tr>
<td>KPRNet</td>
<td>🍀</td>
<td>63.1</td>
<td>✔</td>
<td>✔</td>
</tr>
<tr>
<td>Cylinder3D</td>
<td>🍀</td>
<td>61.8</td>
<td>✔</td>
<td>✔</td>
</tr>
<tr>
<td>FusionNet</td>
<td>🍀</td>
<td>61.3</td>
<td>✔</td>
<td>✔</td>
</tr>
<tr>
<td>SalsaNext</td>
<td>🍀</td>
<td>59.5</td>
<td>✔</td>
<td>✔</td>
</tr>
<tr>
<td>KPConv</td>
<td>🍀</td>
<td>58.8</td>
<td>✔</td>
<td>✔</td>
</tr>
<tr>
<td>SqueezeSegV3</td>
<td>🍀</td>
<td>55.9</td>
<td>✔</td>
<td>✔</td>
</tr>
</tbody>
</table>
Accelerate Video Recognition with TSM

temporal shift module

I3D:
Latency: 164.3 ms/Video  Something-V1 Acc.: 41.6%

TSM:
Latency: 17.4 ms/Video  Something-V1 Acc.: 43.4%

Speed-up: 9x

TSM [Lin et al., ICCV 2019]
Accelerate Video Recognition with TSM

temporal shift module

I3D:
Throughput: 6.1 video/s
Something-V1 Acc.: 41.6%

TSM:
Throughput: 77.4 video/s
Something-V1 Acc.: 43.4%

12.7x larger throughput

TSM [Lin et al., ICCV 2019]
GAN Compression

Accelerating Horse2zebra by GAN Compression

Original CycleGAN; FLOPs: 56.8G; FPS: 12.1; FID: 61.5

GAN Compression; FLOPs: 3.50G (16.2x); FPS: 40.0 (3.3x); FID: 53.6

Measured on NVIDIA Jetson Xavier GPU. Lower FID indicates better performance.
Accelerate Natural Language Processing

Efficient NLP on mobile devices enable real time conversation between speakers using different languages.

“Nice to meet you”

“Encantada de conocerte”

“한나서 반갑습니다”

“很高兴见到你”

“Freut mich, dich kennenzulernen”

---

Human Life (Avg. 1 year) 11,023
American Life (Avg. 1 year) 36,156
US Car w/ Fuel (Avg. 1 lifetime) 126,000
Evolved Transformer
HAT (Ours) 626,155

---

BLEU Score

- HAT (Ours)
- Layer Number Scaling of Transformer
- Dimension Scaling of Transformer

3.0x Faster
3.7x Smaller

Raspberry Pi ARM CPU latency (s)

WMT ’14 En-De

HAT, ACL’20
SpAtten: Efficient Sparse Attention Architecture

- **Motivation:** Attention layer in natural language processing models is the bottleneck for end-to-end performance.
- **Main idea:** Reduce the redundant computation.
  1. **Cascade Token and head pruning:** Based on attention distribution, we remove unimportant tokens and heads to reduce computation and memory access.
  2. **Progressive quantization:** progressively fetch MSB and LSB to reduce average bitwidth. If attention distribution is flat, using MSB is sufficient for accuracy.

**Cascade token and head pruning**

As a visual treat, the film is almost perfect.

- 11 Tokens ↓ 12 Heads

BERT Layer 1 (100% Computation & Memory Access)

As a treat, film perfect.

- 5 Tokens ↓ 10 Heads

Layer 2 (38%)

Cascade pruning of unimportant tokens & heads on the fly.
Not affecting accuracy.

Film perfect

- 2 Tokens ↓ 8 Heads

Layer 3 (12%)

Sentiment Classification: Positive ✓

[HPCA'21] Hanrui Wang, Zhekai Zhang, Song Han; “SpAtten: Efficient Sparse Attention Architecture with Cascade Token and Head Pruning”
SpAtten: Efficient Sparse Attention Architecture

<table>
<thead>
<tr>
<th>Platform</th>
<th>Power (W)</th>
<th>Performance (GFLOPS)</th>
<th>Energy Efficiency (GFLOP/J)</th>
</tr>
</thead>
<tbody>
<tr>
<td>A3 (ASIC)</td>
<td>2.00</td>
<td>221 (1.6x)</td>
<td>269 (1.4x)</td>
</tr>
<tr>
<td>MNNFast (ASIC)</td>
<td>0.823</td>
<td>120 (3.0x)</td>
<td>120 (3.2x)</td>
</tr>
<tr>
<td>SpAtten (ASIC)</td>
<td>0.942</td>
<td>360</td>
<td>382</td>
</tr>
</tbody>
</table>

[HPCA'21] Hanrui Wang, Zhekai Zhang, Song Han; “SpAtten: Efficient Sparse Attention Architecture with Cascade Token and Head Pruning”
Efficient Deep Learning

less computation → fewer engineers

less data
From Manual Design to Automatic Design

Manual Architecture Search
- AlexNet, VGGNet, ResNet, MobileNet...
- Pruning, Quantization, Compression...

Automatic Architecture Search
- Use Machine Learning
- AutoML
AMC: AutoML for Model Compression
[ECCV 2018]

Proxyless Neural Architecture Search
[ICLR 2019]

HAQ: Hardware-aware Automated Quantization
[CVPR 2019], oral
AutoML Consistently Outperforms Human Baselines

- Once-for-all sets a new state-of-the-art 80% ImageNet top-1 accuracy under the mobile vision setting (< 600M MACs).

Once-for-All, ICLR’20
AutoML Consistently Outperforms Human Baselines

- **First place**, 5th IEEE Low-Power Computer Vision Challenge, CPU detection track & FPGA track, Aug 2020 [OFA]
- **First place**, 3D semantic segmentation on SemanticKitti, July 2020 [SPVNAS]
- **First place**, 4th IEEE Low-Power Computer Vision Challenge, CPU classification and detection track, Jan 2020 [OFA]
- **First place**, 3rd IEEE Low-Power Computer Vision Challenge, DSP track, @ICCV 2019 [OFA]
- **First place**, MicroNet Challenge, NLP track (WikiText-103), @NeurIPS 2019 [paper]
- **First place**, Visual Wake Words Challenge, TF-lite track, @CVPR 2019 [ProxylessNAS] [demos]
Fast Specialization on Diverse Hardware Platforms

![Graphs showing Top-1 ImageNet Acc (%) vs. Latency for different hardware platforms: Samsung S7 Edge, Google Pixel2, LG G8, NVIDIA 1080Ti, Intel Xeon CPU, and Xilinx ZU3EG FPGA.](image)
Before Acceleration

CARN 16FPS PNSR:20.94
After Acceleration

Ours 41FPS PNSR:20.69
ProxylessNAS in Industry

- Amazon: landed in AutoGluon [1]
- Facebook: landed in PytorchHub [2]

Efficient Deep Learning

- less computation
- fewer engineers
- less data
Data Is Expensive

FFHQ dataset: \(70,000\) selective post-processed human faces

ImageNet dataset: \(\text{millions}\) of images from diverse categories

\textit{Months or even years} to collect the data, along with \textit{prohibitive} annotation costs.

“The aim of the new directorate is to support fundamental scientific research — with specific goals in mind. This is not about solving incremental technical problems. As one example, in artificial intelligence, the focus would not be on further refining current algorithms, but rather on developing profoundly new approaches that would enable machines to "learn" using much smaller data sets — a fundamental advance that would eliminate the need to access immense data sets.”

— L. Rafael Reif
Baseline Results

![Graph showing FID results for StyleGAN2 (baseline) with different data percentages. The graph indicates that worse quality is given less training data.](image-url)
Our Results

- StyleGAN2 (baseline)
- + DiffAugment (ours)

For 100% data, the FID is 11.1 for StyleGAN2 and 9.9 for + DiffAugment.
For 20% data, the FID is 23.1 for StyleGAN2 and 12.2 for + DiffAugment.
For 10% data, the FID is 36.0 for StyleGAN2 and 14.5 for + DiffAugment.

- Worse quality given less training data.
- Matches state of the art with only 20% data.
Fine-Tuning (70k Images) vs. Ours (100 Images)

- Scale/Shift (Noguchi et al.)
- MineGAN (Wang et al.)
- TransferGAN (Wang et al.)
- FreezeD (Mo et al.)
- Ours

# Training Images

- No pre-training

Data
- 100000
- 10000
- 1000
- 100
- 10
- 1

Performance
- 51
- 38.25
- 25.5
- 12.75
- 0

100-shot Obama
Baseline

Obama
100 images

Cat
160 images

Dog
389 images

StyleGAN2 (baseline)

Ours

StyleGAN2 + DiffAugment (ours)
100-Shot Interpolation

Our code, datasets, and models are publicly available at https://github.com/mit-han-lab/data-efficient-gans.
Reducing the carbon footprint of artificial intelligence

MIT system cuts the energy required for training and running neural networks.

MIT aims for energy efficiency in AI model training

ENN ORIGINAL

Reducing The Carbon Footprint Of Artificial Intelligence

MASSACHUSETTS INSTITUTE OF TECHNOLOGY / 24 APRIL 2020
Pruning

Potential product impact for NVIDIA: future TensorRT and cuDNN libraries.

This is a large design space that's hard to be explored by human. It should be explored by AI. I plan to use machine learning techniques to find the best number representation for deep learning. The design space includes:

- [weight, activation, gradient] x
- [linear quantization, log quantization, kmeans quantization] x

For training, conventional fp16 or fp32 are also efficient, since training DNNs needs more dynamic range and exciting methods need every time) but flexible expressiveness. The latter has ine...
References

Model Compression & NAS
- **Once-For-All**: Train One Network and Specialize It for Efficient Deployment, ICLR’20
- **ProxylessNAS**: Direct Neural Architecture Search on Target Task and Hardware, ICLR’19
- **APQ**: Joint Search for Network Architecture, Pruning and Quantization Policy, CVPR’20
- **HAQ**: Hardware-Aware Automated Quantization with Mixed Precision, CVPR’19, oral
- **Defensive Quantization**: When Efficiency Meets Robustness, ICLR’19
- **AMC**: AutoML for Model Compression and Acceleration on Mobile Devices, ECCV’18

Efficient Vision:
- **GAN Compression**: Learning Efficient Architectures for Conditional GANs, CVPR’20
- Video: **TSM**: Temporal Shift Module for Efficient Video Understanding, ICCV’19
- Point cloud: **PVCNN**: Point Voxel CNN for Efficient 3D Deep Learning, NeurIPS’19, spotlight

Efficient NLP:
- **Lite Transformer** with Long Short Term Attention, ICLR’20
- **HAT**: Hardware-aware Transformer, ACL’20

Hardware & EDA:
- **SpArch**: Efficient Architecture for Sparse Matrix Multiplication, HPCA’20
- **SpAtten**: Efficient Sparse Attention Architecture with Cascade Token and Head Pruning, HPCA’21
- **Transferable Transistor Sizing** with Graph Neural Networks and Reinforcement Learning, DAC’20