# Benchmarking Neural Network Training Algorithms ELLIS Workshop

### **Frank Schneider**

July 19, 2023





Federal Ministry of Education and Research



# Neural Networks are extremely useful models...

In practice, neural networks are...

...but!

- ▶ ...**slow** to train (training can easily take days or weeks)
- ► ...tedious (demanding human intervention and fiddeling)
- ► ...expensive (requiring dozens of trial runs)

### Neural Networks are extremely useful models...

In practice, neural networks are...

...but!

- ▶ ...**slow** to train (training can easily take days or weeks)
- ▶ ...tedious (demanding human intervention and fiddeling)
- ► ...expensive (requiring dozens of trial runs)

We all want neural network training to be faster, more automatic, and more efficient!

# The Three Pillars of Neural Network Training

Our focus is on the (training) algorithms

#### Hardware

- Leveraging accelerators (e.g. GPUs, TPUs, etc.)
- Maximizing accelerator utilization throughout training
- Reducing hardware bottlenecks

#### Software

- Convenient deep learning frameworks (e.g. РуТогсн, JAX, etc.)
- Efficient software implementations
- Putting ML models into production

#### Algorithms

- Efficient training algorithms (e.g. ADAM, SHAMPOO, etc.)
- Powerful deep learning models (e.g. TRANSFORMERS, etc.)
- ► Faster tuning methods (e.g. BayesOpt, etc.)

### The State of Deep Learning Training Methods

A confusingly crowded field of methods & hyperparameters

#### A huge number of training methods...

Naw	Ref.	Name	Rei.	Kome	Ref.
Acceletated	104	C-dDAM	TRUE .	PHLE.	0.44
ACCIE	1219	CADA	104	104.	12.1
AshaMar	11/12	Cool Manageminant	101:5	Prob. Kelmen	124
Artidianti	178	CProje.	LINE .	Tobal	1241
Address Adulters 18	11	Carnellall	10.0	Francistan/ Francistan	1221
Addition	10.0	Distant	1214	Protobilistic Polyaie	1214
Actualities	10.04	Teophteoux	1111	Problé	1211
Ashillenevi	114	DCISiOpé	101	Philade and	104
AddComp	E I R	ChRCrowl	TTT	DELLARS OF DEL	0.19/1
Actacletts	114	Exdans	104	BAclare .	1111
Addition	10.01	ERENC	1-4	Rocger.	110
Achallie	11-8	A	LOC .	Rongeriant	11-01
Actidiant	1110	Poportuned .	1.1	RM0/Prog	12-1
Address	1121	First As failed	L'MAC	#MPRover.	10.0
Adapted	1.14	TRAD	1121	5-9CD	1244
ADAHERIDAN	100	G-Add-Cand	Diff.	SAchter	1111
Ache	15116	CROAM	1001	Andrew, NAMES Lond	(Crief
Addiana	12148	Galim	DOT	SALIE	1734
Adam	Arrest	COALS.	LOI.	SAM	Ref.
Adata"	1110	00654	10 P	AC Adapted SC Rhill Proc.	100
Asheadd.	1.500	Grad-Ana	12.02	AD Pres.	1.61
Actuality	11ml	CHAPTE .	1.1	90.00	1111
Actore195	0.18	Gravalizes	DATE	3025-84	12/11
AdaptNC	10.00	Gravity	1771	SCD-G2	177
Actualities	114	H Sultant	LUNC S	MODEM -	1244
AsheniP (BCIDP)	10.0	Manage Andrews	6.9	ALCO Inc.	1.044
Advent	1008	KOPED-MARCHED	Diff	SCIENT.	100
Actionality	1.1	AT CRUCHN.	1.0	9.10	
A-Tarali.	1000	ADM.	1000	Si Defenseri	1111
AUMA	114	AD LAURINA.	11.1	However,	10,100
A-545	1111	1.1.5-ham/1.0Memories	CHE .	Thursday Section 1.4	1111
Arbeitenter	1111	LAME	TIDE	Surrik D	111
AGMERT	1110	Lifter	11.1	SECOL/ SALINE	1000
Activity 8	10.00	LAR	1110	1043	111
Addinat	1111	LINCOVE	1.1	(shari)	1551
Achietest	1218	Louid About	Dist.	SNK2M	LTC1
Addr. (Addr. WW)	1110	MISTAC	1.4	Solt-Adam	10.10
ABOD	10.00	MADCRAD	1.10	189620	1541
AGE	11/2	5555	1010	they filmed \$211	1.1
Aspellowed	1111	MUCH.	100	MARTIN	
AMPROVAL	121.04	MTAGen	110.0	SAULTIN	11/1
Acutabetical	10.00	MARKED MARKED	1.10	The fairs	1.5
Advanted 10	1000	bind into	100	TENTAL	Doct.
Allerat	110	NAMES STREET	100	Not design	1.1
ACAM	1.1	NTL Adver	100	WENTE	1000
Acad \$5	100	News	Dist.	PERMIT	100
Acathiat	16.2	Melaboria	1111	VALUE AN	100
Ballan.	1000	Name Adam (Noise MillAC)	0.0	Minifiand	10.0
#Chden	111	Nucleise.	Long L	Station No.	1244
IPCrof.	100	Neuropead	101	Madel	1000
REATION	11	NUMBER	and a		
#1677b	1.0	Phillips	1.4		
	1.411.5				

from "Descending through a Crowded Valley" (Schmidt, Schneider, Hennig; 2021)

# The State of Deep Learning Training Methods

A confusingly crowded field of methods & hyperparameters

### A huge number of training methods...

Nam	Ref.	Name	Red.	Kome	Bell.
Accelerated	104	C-4DAM	TRUE .	PHILE.	044
ACCIER	1210	CADA	LOK-	751.	12-1
Adaliller	1.121	Court Microsoft and	101	Paris Asiana	123-1
Actidiates	LTR.	CPreds.	C HC	To-burk	12+1
Addition/Addition/68	11	Cartoball	10.0	Freezint20/Freezint2064	1223
Addition	10.0	Distants	1211	Protobilistic Publishe	121.0
Actualities	10.04	DeepMensors	1111	Problé	1211
Addinari	1140	DCISiOpt	1111	Pinese:	1044
Add Comp	E I R	ChRCroft	TT.	OPERATORIA COMPACT	1101
Actachetta	Diff	Exden	104	B.Aclare	1111
Addition	1041	ERENC.	1-4	Rorger.	1100
Ashelin	11.0	hee	LIGH C	Record and	11-11
Autobiore	1112	Experiment	111	RMD Prop	121-1
Address	1223	From Judia Bellind	TTNE:	RMPROPERTY.	10.0
Admini	1.34	THREE	1121	5-9CD	1. Contraction
ADADEBIGAN	1111	G-Add and	111	SActain	1111
Arba	1.0.00	CROAM .	Contraction of the	Judger NAME and	Contract of Contract
Add in	1110	Calum	1111	5518	1000
Actava	1.1	0041.0	LO.	SAM	Res.
Aritana"	1.00	000.54	100	White and White and	1.00
Admit	1000	Grad day	1111	(Chan	1.00
Actual on	1 mil	CHAPPEN .	1.1	10.11	100.0
Acherolite		Constant		10.00.00	
A-handbar	1000	Crades	100	10.00.000	17.78
4.1.1.1.1.1.1	100	Life from	1.1	NORTH A	1000
Ashend (BC DR)	1.1	Margan b down	6.0	N. T. Balance	1000
A-Dest T		NUMBER OF THE OWNER OF T		10.004	1000
Parameter	100	POR LEVE BRUCKLY		10.000	
PALMETTY .	1.1	No. 4,014. (814		and the second s	
-Schellers	122	Parent.	1.1	St Collegend	1.00
ALMAN .	1.04	PP18/RPSA	1.4	The second second	100,000
noe		Laware (Laware or all the	CT I	affactoration	
ACCEVED.		1.000		The second secon	
Autoral A	1.11	reach	100	394,547,540,14	1000
PAGE NO.	1.1	LAB	1.1	204.5	
Accession .	1.1	110.01		and a state	1000
Acturban	10.0	Lock Abund	10.4	SNOM	1101
Action (Add 8-04)	1000	55 STAC	1.4	SoltAdam	
ABLD	10.00	PRATICIPAL D	1.1	30642D	12-12
ALLES	1.1.1.1	55.58	10.24	Hop Trained St. 10	1.4
AMERICAN	1.44	MERCK	110	996.01%	100
AMOURA	12.78	NO ALIES	111	SWHEE	100
Angebrund	1000	SIV R USTUR	12.4	- Swdaey	1204.1
Arvaip4.3	1000	Paind area	104	TERFAC	191
ABHI	1.0	NAMER/SUMMG	0.0	N/Adorect	104
ASAM	1.1	NO-Mary	1111	AF 90.0	7. 200
Ac6:4.85	1100	Newsy	THE R. L.	198204/196204/199204	1.14
Anacarad	1298	Notketor	1224	HID-M	1.1211
B.R.d.tair	The state	Noticy Addam; Notics IOPAC	10.14	WeiGood	12/1
Richten	1110	Nochdem	THE .	30(Brier 21)	12+0
IlGent	1214	Newcagned	121	Visi	1344
BRASPICE		NE SGH	100		
BAD .	10.00	Philary	1.1		

from "Descending through a Crowded Valley" (Schmidt, Schneider, Hennig; 2021)

#### ...and training tricks

- OneCycle scheduler, gradient checkpointing
- Genetic Algorithm for Hyperparameters
- Avoid batches that lead to NaN/inf losses
- One cycle, low fidelity training, SGD with restarts
- Proximal optimization for regularizers
- Line searches for the maximum learning rate
- Normalized updates
- Distributed Shampoo, Normformer, GLU
- Weight averaging

from the NeurIPS "HITY Workshop" (Schneider et al.; 2022)

- FreezeOut
- A different epsilon value!
- Check hyperparameter performance over multiple seeds
- Lowering the learning rate!
- Normalizing data works better than batch or layer norm
- Mixed precision training
- Train with a small subset
- Cyclic and one cycle LR
- Label smoothing

►

## The Current Benchmarks for Training Algorithms are Insufficient

and this holds back the entire field

### Unclear SOTA

There is no established protocol to train neural networks.

### ► No reliable way to detect algorithmic improvements

Let alone understand what novel methods are most promising.

#### Training algorithms are assumed to not be useful until widely adopted This chicken-and-egg problem is troubling for practitioners training neural networks and developers of new training algorithms.

## The Current Benchmarks for Training Algorithms are Insufficient

and this holds back the entire field

#### Unclear SOTA

There is no established protocol to train neural networks.

### ► No reliable way to detect algorithmic improvements

Let alone understand what novel methods are most promising.

#### Training algorithms are assumed to not be useful until widely adopted This chicken-and-egg problem is troubling for practitioners training neural networks and developers of new training algorithms.

We desperately need new benchmarks for neural network training algorithms!

### MLCommons Introduces the AlgoPef: Training Algorithm Benchmark

An unprecedented effort to find faster deep learning training algorithms

# ML Commons Algorithms Working Group



George Dahl Google



Frank Schneider University of Tübingen

AlgoPerf: Training Algorithm Benchmark A competition to measure neural network training speedups due to algorithmic changes.

- A competitive benchmark with open submissions
- Compete on time-to-result over multiple deep learning workloads
- A huge large-scale effort by 25+ researchers from Google, University of Tübingen, University of Toronto, Meta AI, etc.

# Challenge I: What is the target?

Which algorithms trains the fastest depends on what it means for training to be complete





Figure 1: *Left*: Validation error for two different runs (—, —) of ADAM on RESNET-50 on IMAGENET. *Right*: The *best* validation error obtained so far. The runs intersect multiple times (♥).

# Challenge I: What is the target?

Which algorithms trains the fastest depends on what it means for training to be complete





Figure 1: *Left*: Validation error for two different runs (-, -) of ADAM on RESNET-50 on IMAGENET. *Right*: The *best* validation error obtained so far. The runs intersect multiple times  $(\cong)$ .

- Directly comparing training curves is ill-posed
- Without defining the target in advance, we can champion any method
- We are effectively moving the goal post after the experiment

UNIVERSIT Tübinge



Setting a common goal post

#### ► Defined Target Performances

We define competitive validation and test targets for each workload that can be reliably achieved with currently popular methods.

#### ► A Time-To-Results Competition

Measure the wall-clock runtime until the algorithm first hits the targets.

### ► Fixed Hardware

Submissions need to innovate on the training algorithm.

### Challenge II: Dependence on the Workload

UNIVERSITÄT TÜBINGEN

Seemingly minor changes to the model can have a large effect on the performance of different training algorithms

Training Algorithm	<b>ResNet-200</b> (standard)
Nesterov	<b>0.2090</b>
AdamW	0.2626

 Table 1: Performance of training methods on different workloads. Shown is

 the best error rate (lower is better) achieved.



Seemingly minor changes to the model can have a large effect on the performance of different training algorithms

Training Algorithm	<b>ResNet-200</b> (standard)	<b>ResNet-200</b> Extra-BN
Nesterov	<b>0.2090</b>	No Feasible Trials
AdamW	0.2626	0.2722

Table 1: Performance of training methods on different workloads. Shown is the best error rate (lower is better) achieved.

### Challenge II: Dependence on the Workload

Seemingly minor changes to the model can have a large effect on the performance of different training algorithms

Training Algorithm	<b>ResNet-200</b> (standard)	<b>ResNet-200</b> Extra-BN
Nesterov	<b>0.2090</b>	No Feasible Trials
AdamW	0.2626	0.2722

 Table 1: Performance of training methods on different workloads. Shown is

 the best error rate (lower is better) achieved.

- How general is a performance improvement?
- Comparison with published results is dangerous
- Important to choose relevant workloads



## Solution II: Evaluate on Multiple Fixed Workloads

Finding general-purpose algorithmic improvements

#### ► Fixed Workloads

We don't allow pipeline changes that are not part of the training algorithm.

#### ► Aggregate across Multiple Workloads

Instead of specialized solutions, we want to find the best general-purpose method.

#### ► Use Randomized Workload Variants

To test the robustness of the methods.

### Solution II: Evaluate on Multiple Fixed Workloads



Finding general-purpose algorithmic improvements

Task	Dataset	Model	Loss	Metric	Validation <b>Target</b>	Test <b>Target</b>	Maximum <b>Runtime</b>
Clickthrough rate prediction	CRITEO 1TB	DLRMsmall	CE	CE	0.123 649	0.126 060	7703
MRI reconstruction	FASTMRI	U-NET	L1	SSIM	0.7344	0.741652	8859
Image classification	IMAGENET	ResNet-50 VIT	CE CE	ER ER	0.225 69 0.226 91	0.3440 0.3481	63 008 77 520
Speech recognition	LIBRISPEECH	Conformer DeepSpeech	CTC CTC	WER WER	0.078 477 0.1162	0.046 973 0.068 093	101 780 92 509
Molecular property prediction	OGBG	GNN	CE	mAP	0.280 98	0.268 729	18 477
Translation	WMT 2016	TRANSFORMER	CE	BLEU	30.8491	30.7219	48 151

Often training algorithms are only templates, not run<u>nable procedures</u>

Search Space	Learning Rate	Weight Decay	$1 - \beta_1$	$\beta_2$
AdamW Narrow	[2e-4, 5e-3]	[2e-2, 0.5]	0.1	0.999
AdamW Broad	[5e-6, 2e-2]	[5e-6, 2.0]	[1e-3, 1.0]	0.999

 Table 2: Hyperparameter search spaces for two algorithms using ADAMW.



Often training algorithms are only templates, not runnable procedures

Search Space	Learning Rate	Weight Decay	$1 - \beta_1$	$\beta_2$
AdamW Narrow	[2e-4, 5e-3]	[2e-2, 0.5]	0.1	0.999
AdamW Broad	[5e-6, 2e-2]	[5e-6, 2.0]	[1e-3, 1.0]	0.999

Table 2: Hyperparameter search spaces for two algorithms using ADAMW.

Workload		AdamW Narrow	AdamW Broad
Criteo 1TB	DLRMsmall	0.124 01	0.124 087
fastMRI	U-NET	0.734746	0.734 311
ImageNet	ResNet-50	0.232 56	0.24334
	VIT	0.219 92	0.23616
LibriSpeech	Conformer	0.075 989	0.080 673
	DeepSpeech	0.112 706	0.120 674
OGBG	GNN	0.28214	0.276 307
WMT 2016	TRANSFORMER	31.3523	30.9950

 Table 3: Performance across multiple workloads for two ADAMW training methods. Bolded number highlights the better performance.



Often training algorithms are only templates, not runnable procedures

Search Space	Learning Rate	Weight Decay	$1 - \beta_1$	$\beta_2$
AdamW Narrow	[2e-4, 5e-3]	[2e-2, 0.5]	0.1	0.999
AdamW Broad	[5e-6, 2e-2]	[5e-6, 2.0]	[1e-3, 1.0]	0.999

 Table 2: Hyperparameter search spaces for two algorithms using ADAMW.

Workload		AdamW Narrow	ADAMW BROAD
Criteo 1TB	DLRMsmall	0.124 01	0.124 087
FASTMRI	U-NET	0.734746	0.734 311
IMAGENET	ResNet-50	0.232 56	0.24334
	VIT	0.219 92	0.23616
LibriSpeech	Conformer	0.075 989	0.080 673
	DeepSpeech	0.112 706	0.120 674
OGBG	GNN	0.28214	0.276 307
WMT 2016	TRANSFORMER	31.3523	30.9950

 Table 3: Performance across multiple workloads for two ADAMW training methods. Bolded number highlights the better performance.

- Methods have hyperparameters that need to be set/tuned
- We can't just ignore them as they are workload-dependent
- Same update rule with different hyperparameters is effectively a different algorithm





# Solution III: Make Tuning an Explicit Part of the Benchmark

Submitting algorithms not templates

eberhard karls UNIVERSITAT TÜBINGEN



► Hyperparameter (Search Spaces) are Part of the Submission Submitters have to provide search spaces or default values.

#### ► A Competitive Benchmark

Everyone submits their algorithm and how it should be tuned, therefore generating strong baselines.

#### ► Two Tuning Rulesets

- ► In the **self-tuning** ruleset, everything is done on the clock (e.g. line-searches, freeze-thaw, default hyperparameters, etc.).
- ▶ The **external** tuning ruleset allows parallel resources and only the fastest trial counts.

### Summary

- ► Neural networks are useful but expensive models
- We are currently unable to identify which training algorithms are "best"
- We created the AlgoPerf: Training Algorithms Benchmark to find faster deep learning training algorithms
  - A competitive, time-to-result benchmark
  - Running on fixed hardware and workloads
  - Computing a joint score across multiple realistic workloads

Read the Rules github.com/mlcommons/algorithmic-efficiency/blob/main/RULES.md

**Read the Paper** 

arxiv.org/abs/2306.07179

Submit!

Call for Submission coming soon