

What do we gain with scale?

Why we are building a ladder
to the moon.

LxMLS 2024

Sara Hooker - Cohere For AI

Cohere For AI

Exploring the unknown, together.

Research

- Research Lab
- Publications
- Scholars Program

Open Science

- Cross-institutional collaborations
- Open science initiatives

Forum

- Fireside Chats
- Technical Talks
- Guest Series
- AI Policy/Safety

Fundamental research on critical areas like efficiency, LLMs at scale, safety, hardware/software interaction.

Metadata Archaeology:

Unearthing Data
Subsets by Leveraging
Training Dynamics



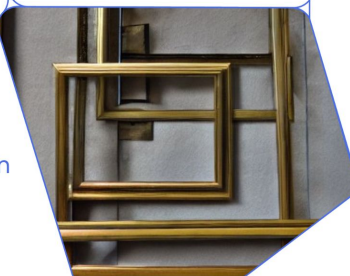
Robust Distillation for Worst-class Performance



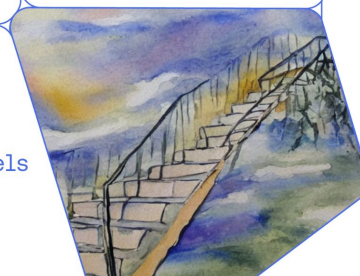
LLMs are not Zero Shot Communicators



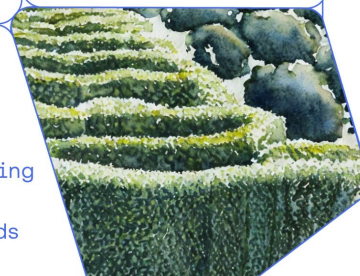
Intriguing Properties of Compression on Multilingual Models



Scalable Training of Language Models using PAX pjit and TPUv4



Studying the impact of magnitude pruning on contrastive learning methods



Our goal is to contribute cutting edge research to the wider ecosystem:

- Research staff published 38 papers last year
- Collaborated across 40+ institutions and organizations
- Released state-of-art models for massively multilingual Aya and C4AI Command-R

✧ Cohere For AI 2023 Research

When Less is More:

Investigating Data
Pruning for Pretraining
LLMs at Scale



The Data Provenance Initiative:

A Large Scale Audit
of Datasets' Licensing
& Attribution in AI



FAIR-Ensemble: When Fairness Naturally Emerges From Deep Ensembling



Evaluating the Social Impact of Generative AI Systems in Systems and Society



The Grand Illusion:

The Myth of Software
Portability and Interoperability
for ML Progress



On the Challenges of Using Black-Box APIs for Toxicity Evaluation in Research



Intriguing Properties of Quantization at Scale



The Presidio Recommendations on Responsible Generative AI



Efficient Methods for Natural Language Processing:

A Survey



Elo Uncovered:

Robustness and Best
Practices in Language
Model Evaluation



Which Prompts Make The Difference?

Data Prioritization
For Efficient Human
LLM Evaluation



Research
support for
compute.

<https://txt.cohere.com/c4ai-research-grants/>

✂ Cohere For AI

Research Grant Program

✂ Cohere For AI



I currently work on designing large scale language models that are **efficient, multilingual, reliable and trustworthy.**

If any of these topics are interesting the talk, happy to discuss after the talk.



Today I will be giving a talk about a paper I recently released with a very boring title, but which grapples with topics that are at the center of computer science progress.

On the Limitations of Compute Thresholds as a Governance Strategy.

Sara Hooker

Abstract

At face value, this essay is about understanding a fairly esoteric governance tool called compute thresholds. However, in order to grapple with whether these thresholds will achieve anything, we must first understand how they came to be. This requires engaging with a decades-old debate at the heart of computer science progress, namely, is “bigger always better?” Hence, this essay may be of interest not only to policymakers and the wider public but also to computer scientists interested in understanding the role of compute in unlocking breakthroughs. Does a certain inflection point of compute result in changes to the risk profile of a model? This discussion is increasingly urgent given the wide adoption of governance approaches that suggest greater compute equates with higher propensity for harm. Several leading frontier AI companies have released *responsible scaling* policies. Both the White House Executive Orders on AI Safety (EO) and the EU AI Act encode the use of FLOP or “floating-point operations” as a way to identify more powerful systems. What is striking about the choice of compute thresholds to-date is that no models currently deployed in the wild fulfill the current criteria set by the EO. This implies that the emphasis is often *not* on auditing the risks and harms incurred by currently deployed models – but rather is based upon the belief that future levels of compute will introduce unforeseen new risks. A key conclusion of this essay is that compute thresholds as currently implemented **are shortsighted and likely to fail to mitigate risk**. Governance that is overly reliant on compute fails to understand that the relationship between compute and risk is highly uncertain and rapidly changing. It also overestimates our ability to predict what abilities emerge at different scales. This essay ends with recommendations for a better way forward.



There are policy implications to this work, but today I will focus on the technical aspects of this paper.

You, see to understand whether compute thresholds make sense – we need to grapple with a question which has been a decades old debate at the heart of computer science progress “is bigger always better?”

On the Limitations of Compute Thresholds as a Governance Strategy.

Sara Hooker

Abstract

At face value, this essay is about understanding a fairly esoteric governance tool called compute thresholds. However, in order to grapple with whether these thresholds will achieve anything, we must first understand how they came to be. This requires engaging with a decades-old debate at the heart of computer science progress, namely, is “bigger always better?” Hence, this essay may be of interest not only to policymakers and the wider public but also to computer scientists interested in understanding the role of compute in unlocking breakthroughs. Does a certain inflection point of compute result in changes to the risk profile of a model? This discussion is increasingly urgent given the wide adoption of governance approaches that suggest greater compute equates with higher propensity for harm. Several leading frontier AI companies have released *responsible scaling* policies. Both the White House Executive Orders on AI Safety (EO) and the EU AI Act encode the use of FLOP or “floating-point operations” as a way to identify more powerful systems. What is striking about the choice of compute thresholds to-date is that no models currently deployed in the wild fulfill the current criteria set by the EO. This implies that the emphasis is often *not* on auditing the risks and harms incurred by currently deployed models – but rather

Today:

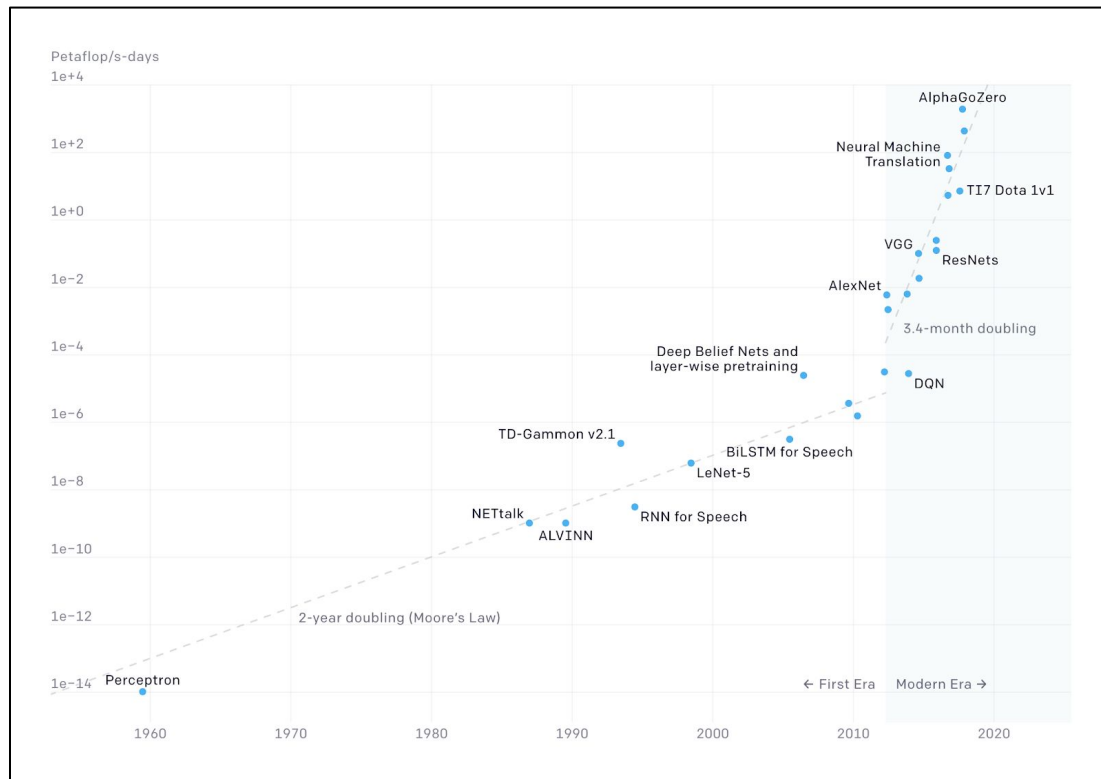
The role of
capacity: Why
we are
building a
ladder to
moon.

Promising
directions of
research

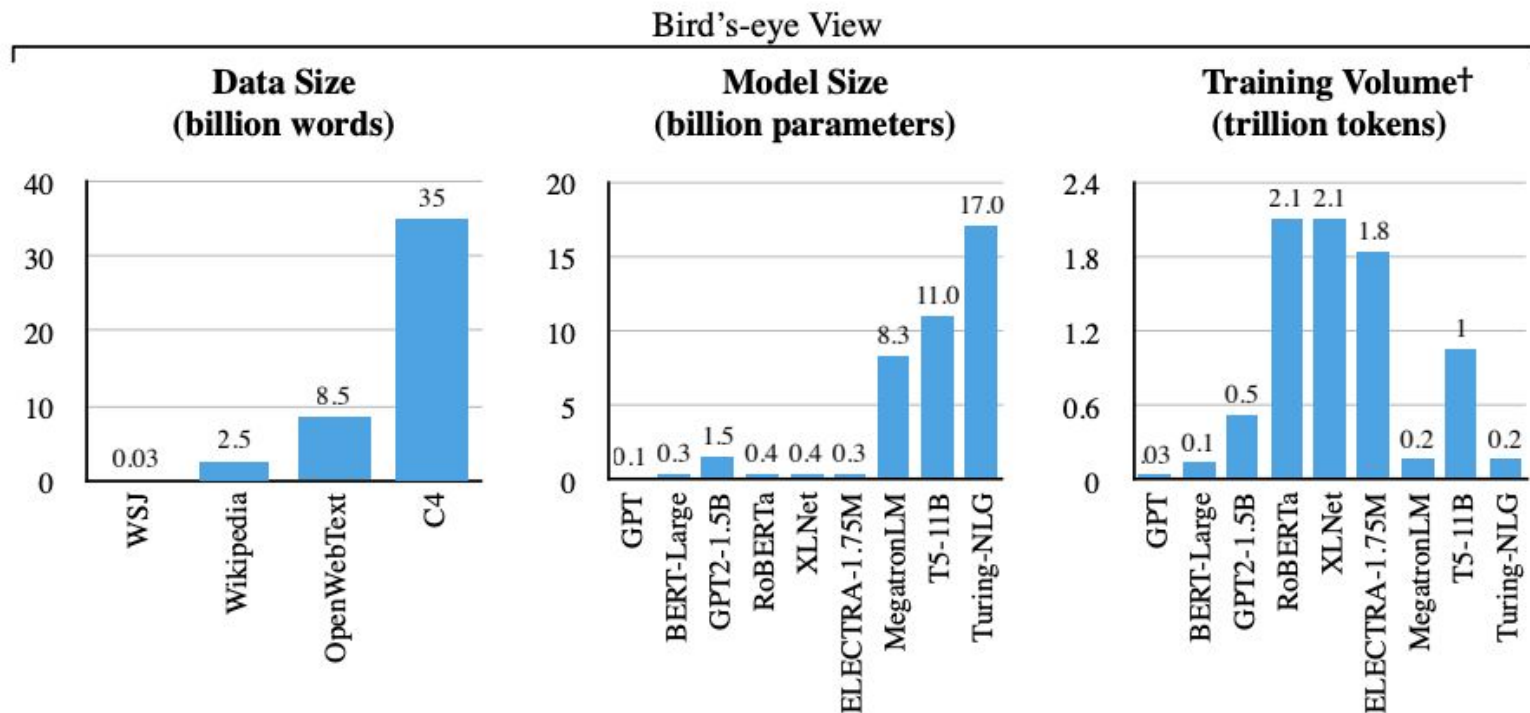
Multilingual –
why it is an
interesting
problem.

The role of model scale and data in recent breakthroughs

A “bigger is better” race in the number of model parameters has gripped the field of machine learning.

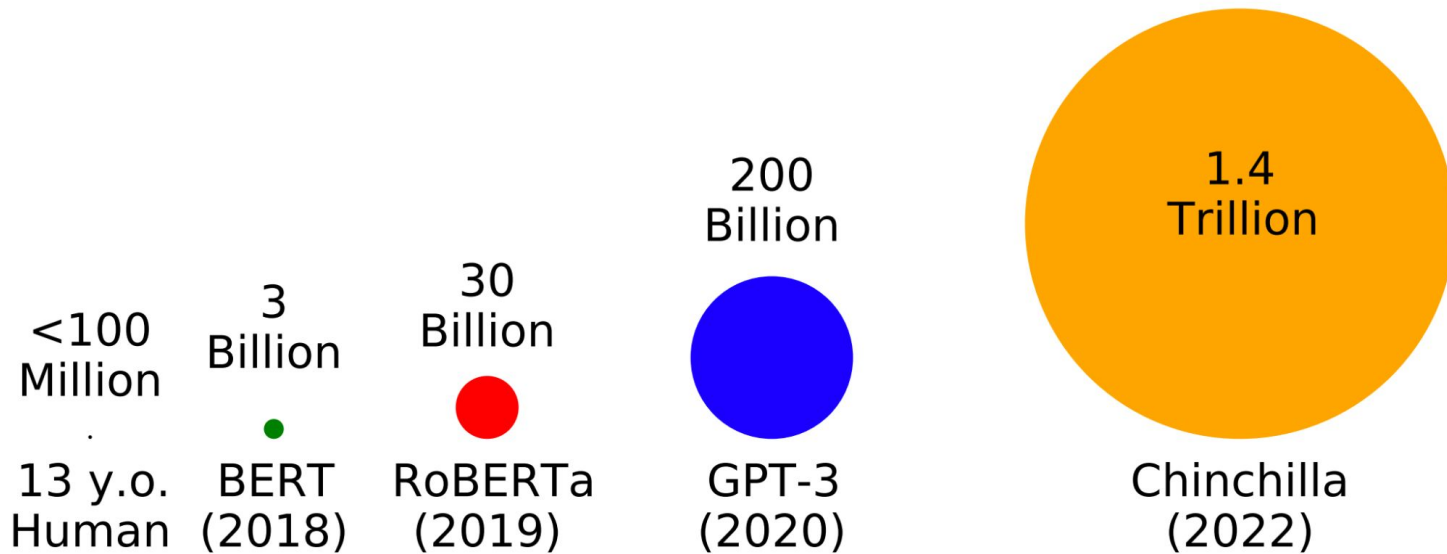


This characterizes both vision and NLP tasks.



Link [here](#) [Sharir et al. 2020]

And involves large increases in both model and dataset sizes:



Number of tokens involved in training.

This has been formalized by Rich Sutton as the “bitter lesson”

The Bitter Lesson

Rich Sutton

March 13, 2019

The biggest lesson that can be read from 70 years of AI research is that general methods that leverage computation are ultimately the most effective, and by a large margin. The ultimate reason for this is Moore's law, or rather its generalization of continued exponentially falling cost per unit of computation. Most AI research has been conducted as if the computation available to the agent were constant (in which case leveraging human knowledge would be one of the only ways to improve performance) but, over a slightly longer time than a typical research project, massively more computation inevitably becomes available. Seeking an improvement that makes a difference in the shorter term, researchers seek to leverage their human knowledge of the domain, but the only thing that matters in the long run is the leveraging of computation. These two need not run counter to each other, but in practice they tend to. Time spent on one is time not spent on the other. There are psychological commitments to investment in one approach or the other. And the human-knowledge approach tends to complicate methods in ways that make them less suited to taking advantage of general methods leveraging computation. There were many examples of AI researchers' belated learning of this bitter lesson, and it is instructive to review some of the most prominent.

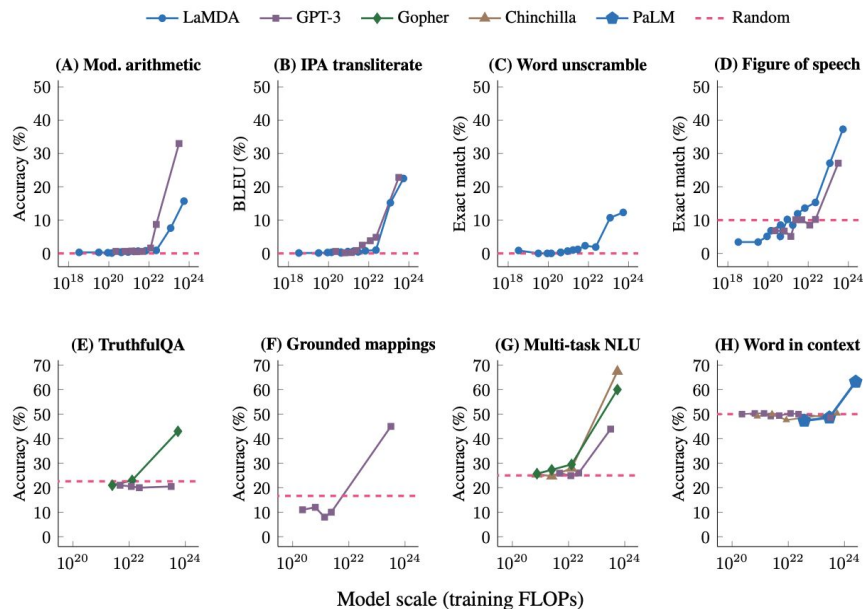
In computer chess, the methods that defeated the world champion, Kasparov, in 1997, were based on massive, deep search. At the time, this was looked upon with dismay by the majority of computer-chess researchers who had pursued methods that leveraged human understanding of the special structure of chess. When a simpler, search-based approach with special hardware and software proved vastly more effective, these human-knowledge-based chess researchers were not good losers. They said that “brute force” search may have won this time, but it was not a general strategy, and anyway it was not how people played chess. These researchers wanted methods based on human input to win and were disappointed when they did not.

A similar pattern of research progress was seen in computer Go, only delayed by a further 20 years. Enormous initial efforts went into avoiding search by taking advantage of human knowledge, or of the special features of the game, but all those efforts proved irrelevant, or worse, once search was applied effectively at scale. Also important was the use of learning by self play to learn a value function (as it was in many other games and even in chess, although learning did not play a big role in the 1997 program that first beat a world champion). Learning by self play, and learning in general, is like search in that it enables massive computation to be brought to bear. Search and learning are the two most important classes of techniques for utilizing massive amounts of computation in AI research. In computer Go, as in computer chess, researchers' initial effort was directed towards utilizing human understanding (so that less search was needed) and only much later was much greater success had by embracing search and learning.

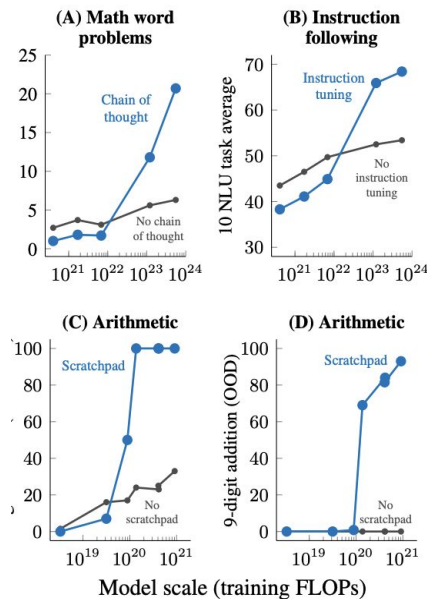
In a punch to the ego of every computer scientist out there, what Sutton is saying is that symbolic methods that codify human knowledge have not worked as well as letting a model learn patterns for itself

Is Sutton right?

Is Sutton right? Different regimes of scale appear to induce emergent abilities.



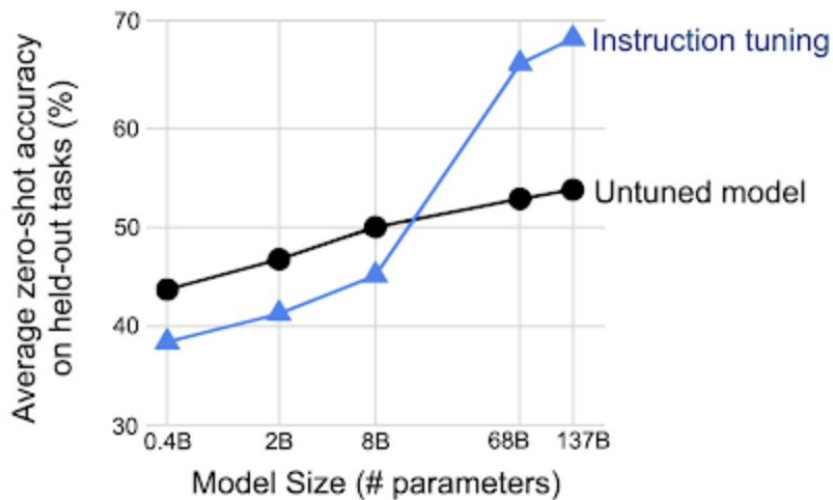
Few shot prompting
performance



Finetuning and few
shot.

[[Wei et al. 2022](#)]

For example, requires larger and larger models to take advantage of instruction fine-tuning.



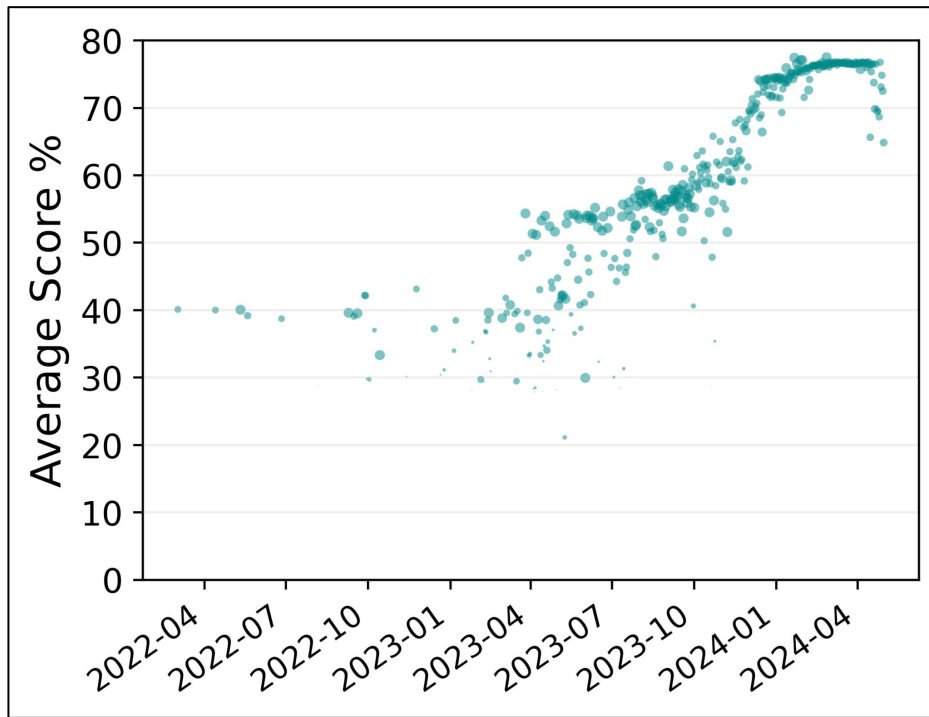
Instruction tuning only improves performance on unseen tasks for models of certain size.

Throwing compute at a problem is still widely favored:

- More de-risked vs more difficult approaches of crafting new optimization techniques
- Fits into industry quarterly planning cycles – hard to justify deviating from the predictable path of gains.

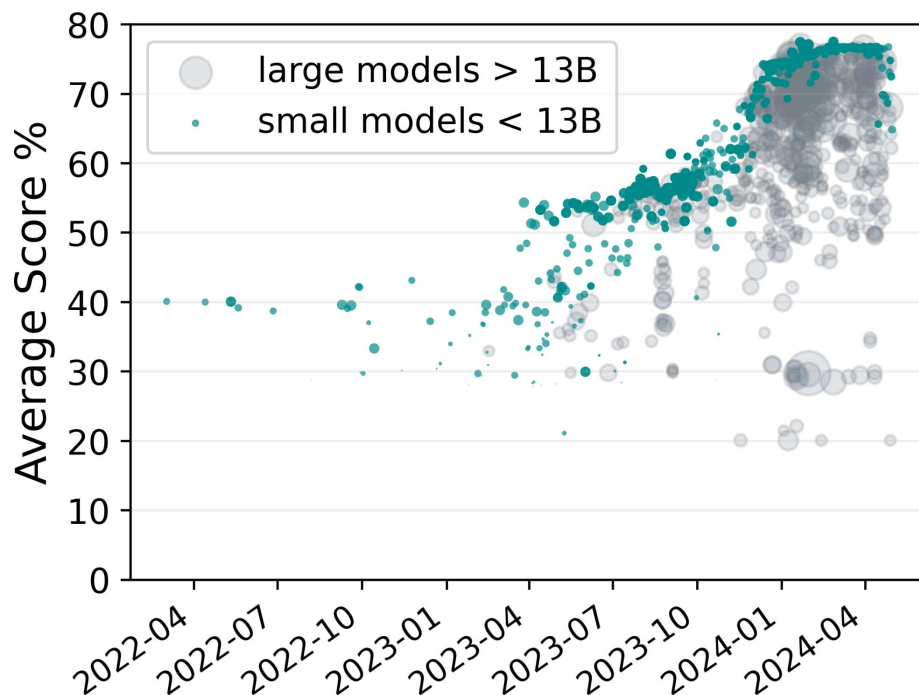
However, many data points exist to suggest that training compute alone is not sufficient.

Models at the same capacity have been getting far more performant over time.

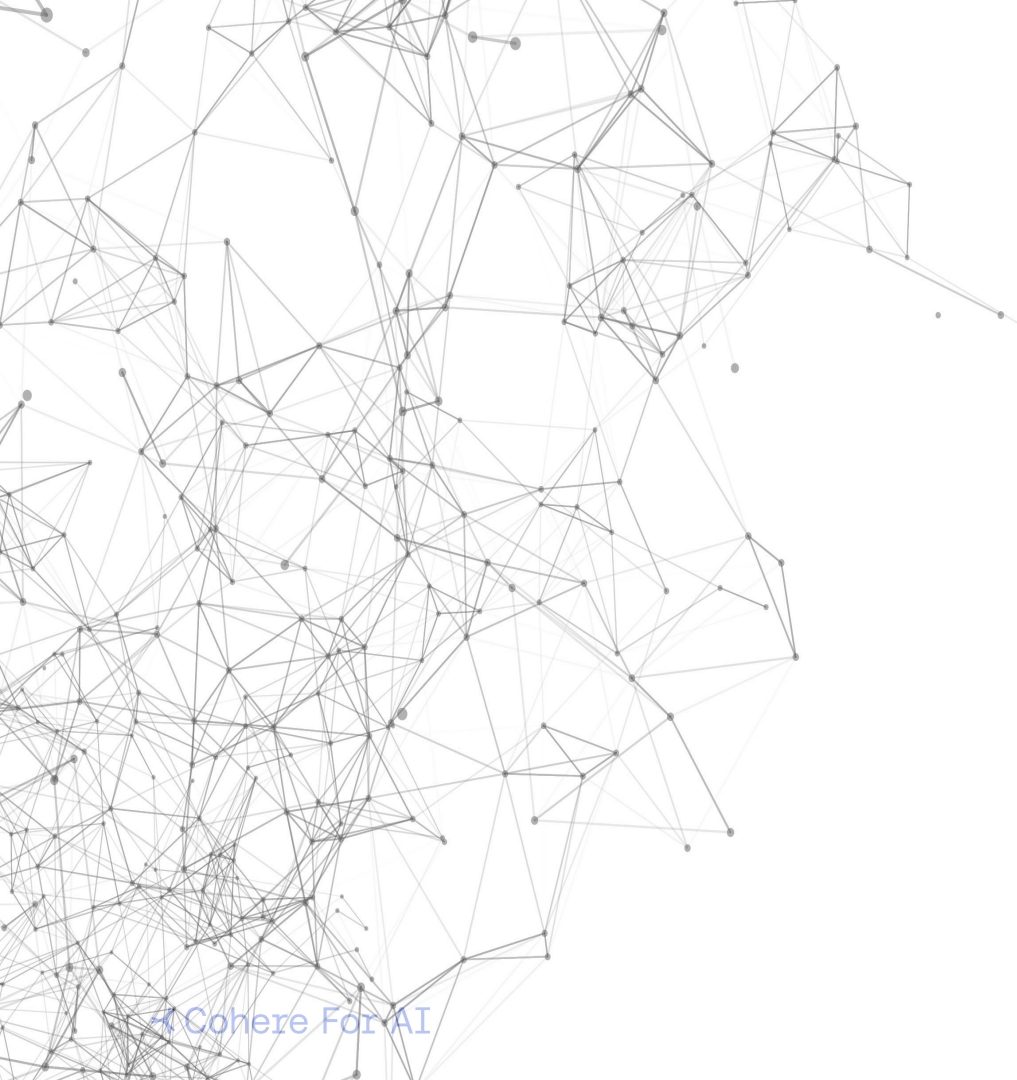


Models **under 13B** on the llm open leaderboard over time.

Smaller models frequently outperform far larger models.



All models **over 13B** (grey) that underperform the best daily model **under 13B** submitted to the llm open leaderboard (green).



A key limitation of blindly
following “the bitter
lesson” is that the
relationship between
compute and
performance properties
is not well understood.

In fact, we observe a highly uncertain relationship between compute and performance.

In fact, we observe a highly uncertain relationship between compute and performance.

- 1) Data quality compensates for need for compute
- 2) Architecture plays a significant role in determining scalability
- 3) Post-training optimization reduces need for training time compute.
- 4) Diminishing returns to adding more weights.
- 5) Many redundancies between weights
- 6) We can remove most weights after training.
- 7) Most weights necessary for representing a small fraction of the

Data quality compensates
for the need for compute.

Recent work finds smaller amounts of higher quality data removes the need for a larger model.

There is increasing evidence that efforts to better curate training corpus, including **deduping, pruning data and increasing the available training corpus size** can compensate for the need for larger networks and/or improve training dynamics.

	% train examples with dup in train	% valid with dup in valid	% valid with dup in train
C4	3.04%	1.59%	4.60%
RealNews	13.63%	1.25%	14.35%
LM1B	4.86%	0.07%	4.92%
Wiki40B	0.39%	0.26%	0.72%

Table 2: The fraction of examples identified by NEARDUP as near-duplicates.

[Lee et al. 2022](#)

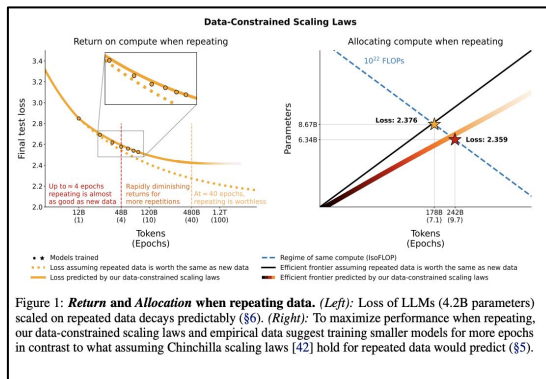


Figure 1: **Return and Allocation when repeating data.** (Left): Loss of LLMs (4.2B parameters) scaled on repeated data decays predictably (§6). (Right): To maximize performance when repeating, our data-constrained scaling laws and empirical data suggest training smaller models for more epochs in contrast to what assuming Chinchilla scaling laws [42] hold for repeated data would predict (§5).

[Muennighoff et al. 2023](#)

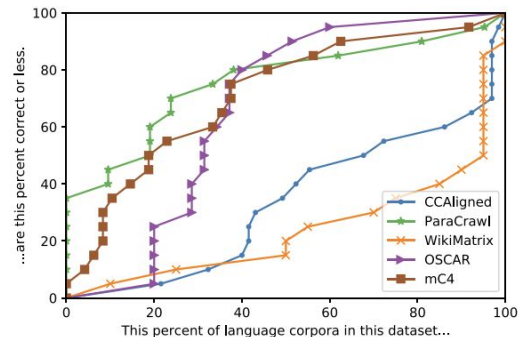
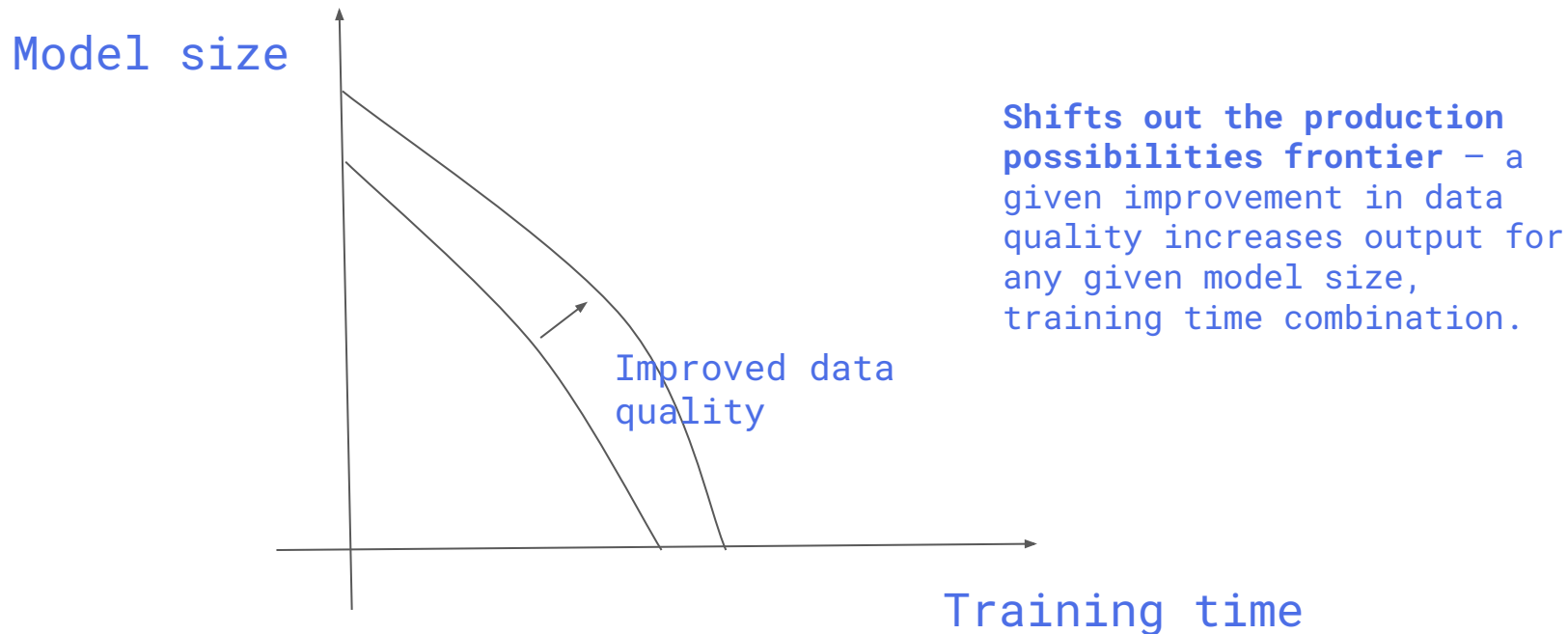


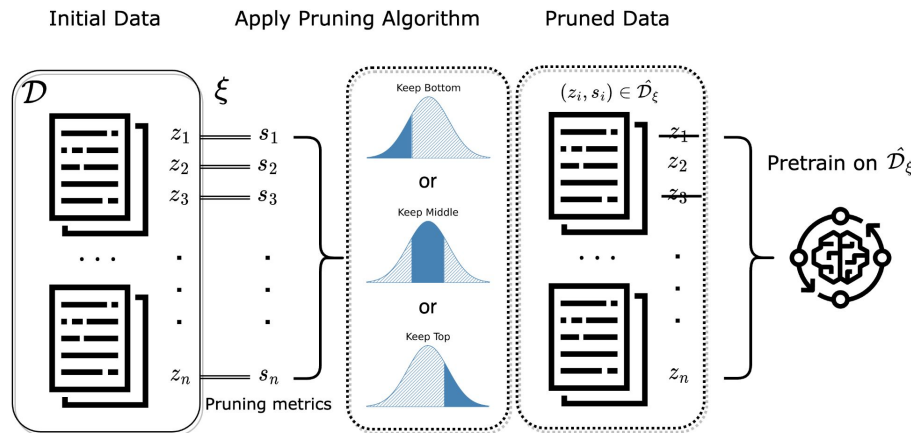
Figure 1: Fraction of languages in each dataset below a given quality threshold (percent correct).

[Kreutzer et al. 2022](#)

This fundamentally changes the notion of the axes of capacity – that instead of just more training time, more weights. That the quality of the data can imply less training time or less weights are needed.



Our recent work focuses on effective data pruning for pretraining internet scale.



When Less is More: Investigating Data Pruning for Pretraining LLMs at Scale

Max Marion
Cohere for AI
maxwell@cohere.com

Ahmet Üstün
Cohere for AI
ahmet@cohere.com

Luiza Pozzobon
Cohere for AI
luiza@cohere.com

Alex Wang
Cohere
alexwang@cohere.com

Marzieh Fadaee
Cohere for AI
marzieh@cohere.com

Sara Hooker
Cohere for AI
sarahooker@cohere.com

Abstract

Large volumes of text data have contributed significantly to the development of large language models (LLMs) in recent years. This data is typically acquired by scraping the internet, leading to pretraining datasets comprised of noisy web text. To date, efforts to prune these datasets down to a higher quality subset have relied on hand-crafted heuristics encoded as rule-based filters. In this work, we take a wider view and explore scalable estimates of data quality that can be used to systematically measure the quality of pretraining data. We perform a rigorous comparison at scale of the simple data quality estimator of perplexity, as well as more sophisticated and computationally intensive estimates of the Error L2-Norm and memorization. These metrics are used to rank and prune pretraining corpora, and we subsequently compare LLMs trained on these pruned datasets. Surprisingly, we find that the simple technique of perplexity outperforms our more computationally expensive scoring methods. We improve over our no-pruning baseline while training on as little as 30% of the original training dataset. Our work sets the foundation for unexplored strategies in automatically curating high quality corpora and suggests the majority of pretraining data can be removed while retaining performance.

We can improve over our no-pruning baseline **while training on as little as 30% of the original training dataset.**

When Less is More: Investigating Data Pruning for Pretraining LLMs at Scale

Max Marion
Cohere for AI
maxwell@cohere.com

Ahmet Üstün
Cohere for AI
ahmet@cohere.com

Luiza Pozzobon
Cohere for AI
luiza@cohere.com

Alex Wang
Cohere
alexwang@cohere.com

Marzieh Fadaee
Cohere for AI
marzieh@cohere.com

Sara Hooker
Cohere for AI
sarahooker@cohere.com

Abstract

Large volumes of text data have contributed significantly to the development of large language models (LLMs) in recent years. This data is typically acquired by scraping the internet, leading to pretraining datasets comprised of noisy web text. To date, efforts to prune these datasets down to a higher quality subset have relied on hand-crafted heuristics encoded as rule-based filters. In this work, we take a wider view and explore scalable estimates of data quality that can be used to systematically measure the quality of pretraining data. We perform a rigorous comparison at scale of the simple data quality estimator of perplexity, as well as more sophisticated and computationally intensive estimates of the Error L2 Norm and memorization. These metrics are used to rank and

[[[Marion et al. 2023](#)]]

Data pruning is a valuable optimization at multiple stages of training pipeline – here we also show promising results in preference training.

We reduce instances of indecisive (or “tie”) outcomes by up to 54% compared to a random sample when focusing on the top-20 percentile of prioritized instances.

This helps save valuable human feedback for the most important instances.

Which Prompts Make The Difference? Data Prioritization For Efficient Human LLM Evaluation

Meriem Boudir
Cohere for AI
meri.boudir@gmail.com

Edward Kim
Cohere
edward@cohere.com

Beyza Ermis
Cohere for AI
beyza@cohere.com

Marzieh Fadaee
Cohere for AI
marzieh@cohere.com

Sara Hooker
Cohere for AI
sarahooker@cohere.com

Abstract

Human evaluation is increasingly critical for assessing large language models, capturing linguistic nuances, and reflecting user preferences more accurately than traditional automated metrics. However, the resource-intensive nature of this type of annotation process poses significant challenges. The key question driving our work: *is it feasible to minimize human-in-the-loop feedback by prioritizing data instances which most effectively distinguish between models?* We evaluate several metric-based methods and find that these metrics enhance the efficiency of human evaluations by minimizing the number of required annotations, thus saving time and cost, while ensuring a robust performance evaluation. We show that our method is effective across widely used model families, reducing instances of indecisive (or “tie”) outcomes by up to 54% compared to a random sample when focusing on the top-20 percentile of prioritized instances. This potential reduction in required human effort positions our approach as a valuable strategy in future large language model evaluations.

Relationship between
weights and performance is
not well understood.

1. **Diminishing returns** to adding parameters. Millions of parameters are needed to **peek** out additional gains.

Model	Parameters ^a	Features	Image Size	Paper	ImageNet Top-1 Accuracy	
					Public Checkpoint ^b	1
Inception v1 ^c [69]	5.6M	1024	224	73.2	69.8	
BN-Inception ^d [34]	10.2M	1024	224	74.8	74.0	
Inception v3 [70]	21.8M	2048	299	78.8	78.0	
Inception v4 [68]	41.1M	1536	299	80.0	80.2	
Inception-ResNet v2 [68]	54.3M	1536	299	80.1	80.4	
ResNet-50 v1 ^e [29, 26, 25]	23.5M	2048	224	76.4	75.2	
ResNet-101 v1 [29, 26, 25]	42.5M	2048	224	77.9	76.4	
ResNet-152 v1 [29, 26, 25]	58.1M	2048	224	N/A	76.8	
DenseNet-121 [31]	7.0M	1024	224	75.0	74.8	
DenseNet-169 [31]	12.5M	1024	224	76.2	76.2	
DenseNet-201 [31]	18.1M	1024	224	77.4	77.3	
MobileNet v1 [30]	3.2M	1024	224	70.6	70.7	
MobileNet v2 [61]	2.2M	1280	224	72.0	71.8	
MobileNet v2 (1.4) [61]	4.3M	1792	224	74.7	75.0	
NASNet-A Mobile [84]	4.2M	1056	224	74.0	74.0	
NASNet-A Large [84]	84.7M	4032	331	82.7	82.7	

Almost double the amount of weights for a gain in 2% points.

Table: [Kornblith et al., 2018](#) [[Kaplan + 2020](#)]

2. Redundancies Between Weights

Predicting Parameters in Deep Learning

Misha Denil¹ Babak Shakibi² Laurent Dinh³
Marc'Aurelio Ranzato⁴ Nando de Freitas^{1,2}

¹University of Oxford, United Kingdom

²University of British Columbia, Canada

³Université de Montréal, Canada

⁴Facebook Inc., USA

{misha.denil,nando.de.freitas}@cs.ox.ac.uk

laurent.dinh@umontreal.ca

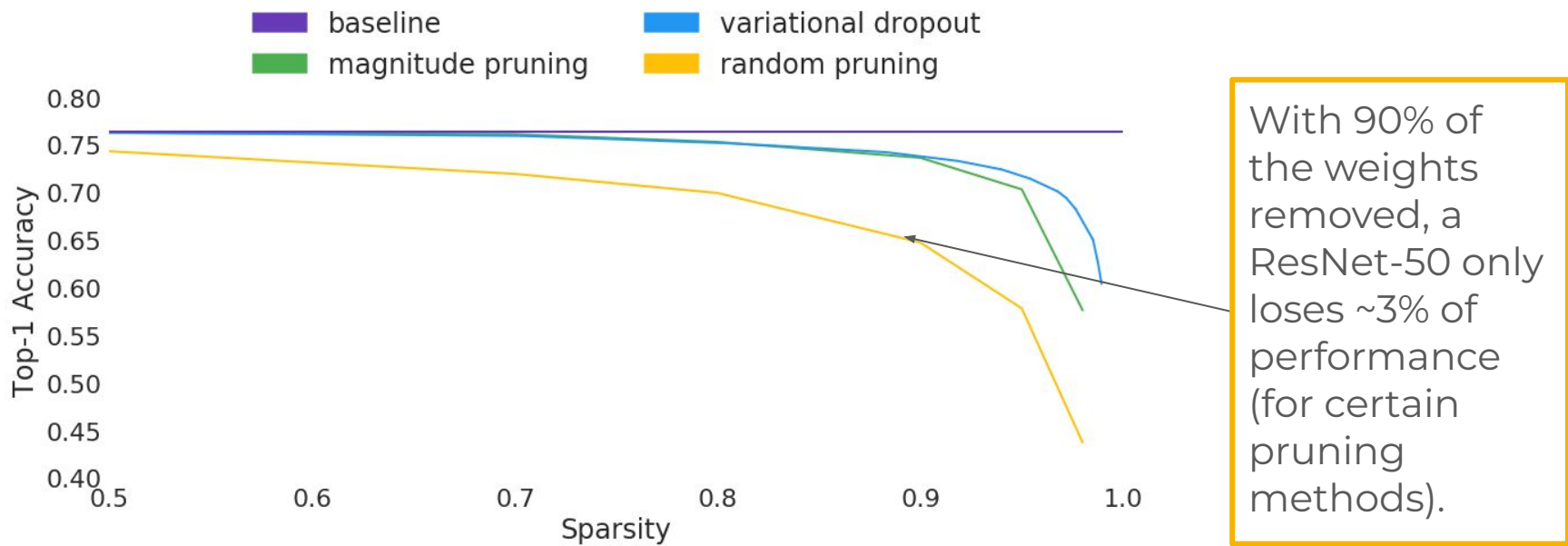
ranzato@fb.com

Abstract

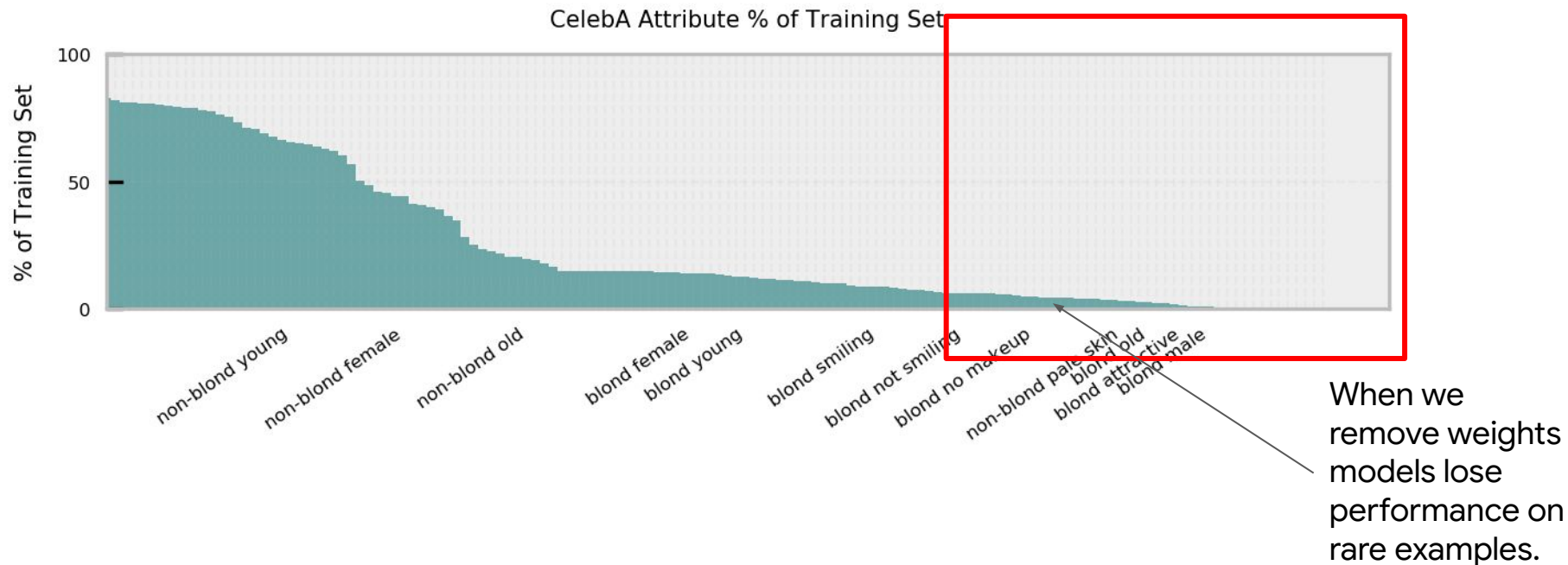
We demonstrate that there is significant redundancy in the parameterization of several deep learning models. Given only a few weight values for each feature it is possible to accurately predict the remaining values. Moreover, we show that not only can the parameter values be predicted, but many of them need not be learned at all. We train several different architectures by learning only a small number of weights and predicting the rest. In the best case we are able to predict more than 95% of the weights of a network without any drop in accuracy.

Denil et al. find that a small set of weights can be used to predict 95% of weights in the network.

3. Most weights can be removed after training is finished (**while only losing a few % in test-set accuracy!**)



4. Across a variety of settings and modalities, we find that removing weights causes models to loss performance on the long-tail. The majority of weights **(90% of all weights)** are used to memorize very rare examples in the dataset.



Compute does not capture
the algorithmic
breakthroughs that have led
to large gains in performance
over last 5 years.

Key algorithmic improvements which have relatively limited compute cost but huge impact on performance:

Key algorithmic improvements which have relatively limited compute cost but huge impact on performance:

1. Multi-task training
2. Instruction fine tuning
3. Preference training

There has been renewed interest in the types of data that lead to powerful zero shot abilities – **performing well on completely unseen tasks.**



Tell me a story about a wizard fish who goes to boarding school underwater.

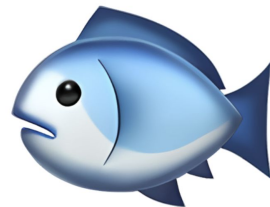


There once was a fish with a special gift,
A wizard in his own right, so wise and deft.
He lived in an underwater school,
Where students gathered from far and wide.

Each day, the wizard fish would cast
A spell to protect the school from harm.
He'd conjure up a shield so strong and vast,
Keeping them all safe from any storm.

But one day, a fierce shark appeared,
With an army of evil fish, they'd been hired.
They came to destroy the wizard fish's school,
And take over the underwater rule.

The wizard fish knew he had to act,
So he gathered his friends, ready for the attack.
With his magic, he created a plan,
To outsmart the shark and win the game.



Two ingredients have been particularly important at leading to breakthroughs in performance on zero and few shot tasks:

**1. Instruction tuning –
Structuring multi-task
fine-tuning data as
questions and
answers**

**2. Integrating human
feedback about
preferences**

Multi-task fine-tuning moves to a single global model – train on multiple tasks at once.

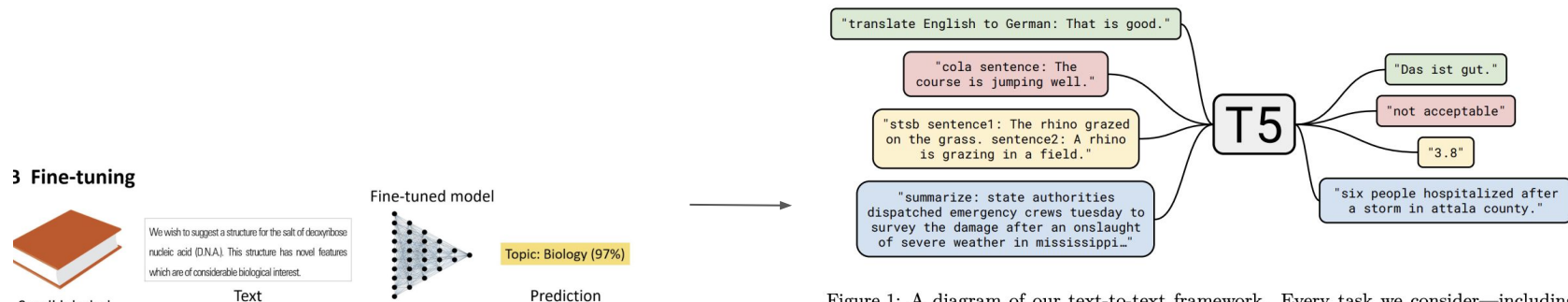


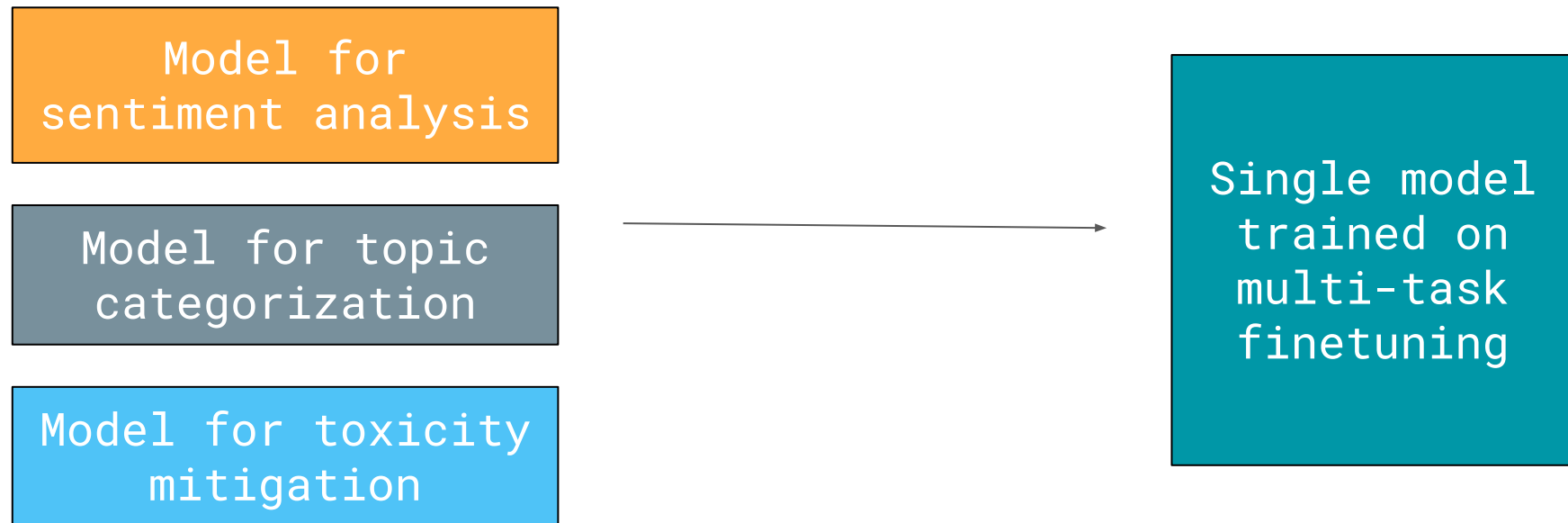
Figure 1: A diagram of our text-to-text framework. Every task we consider—including translation, question answering, and classification—is cast as feeding our model text as input and training it to generate some target text. This allows us to use the same model, loss function, hyperparameters, etc. across our diverse set of tasks. It also provides a standard testbed for the methods included in our empirical survey. “T5” refers to our model, which we dub the “Text-to-Text Transfer Transformer”.

Finetuning on a single task



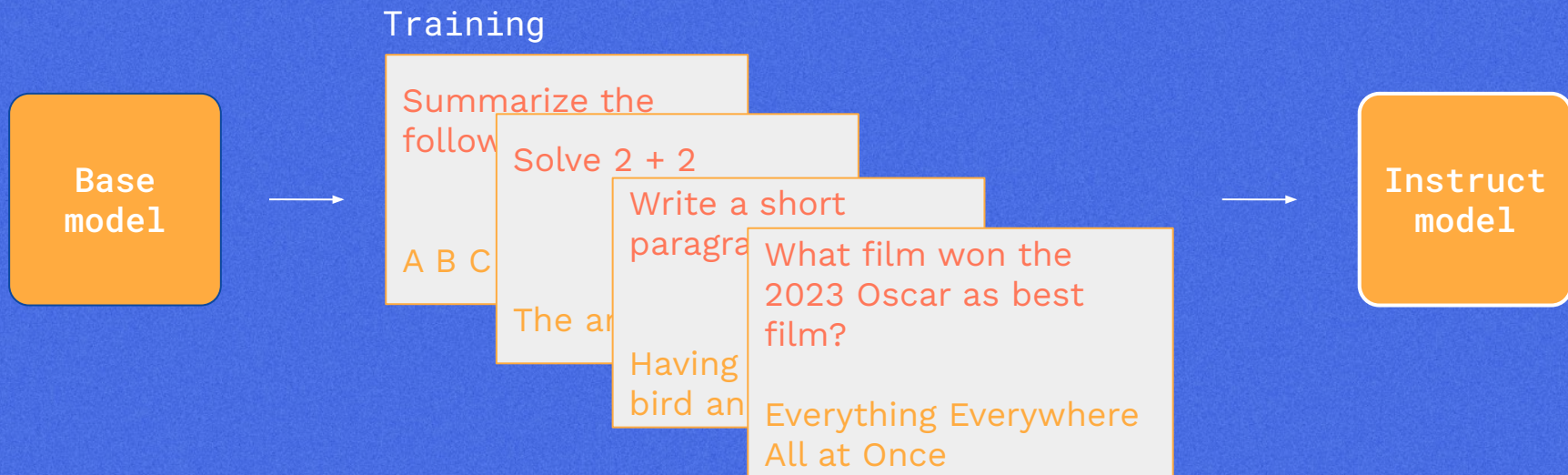
Finetuning on many different tasks

Why is this a big deal – it transitions from having custom models for each task to having a single task-general model that can perform a lot of tasks, which only require zero or few examples

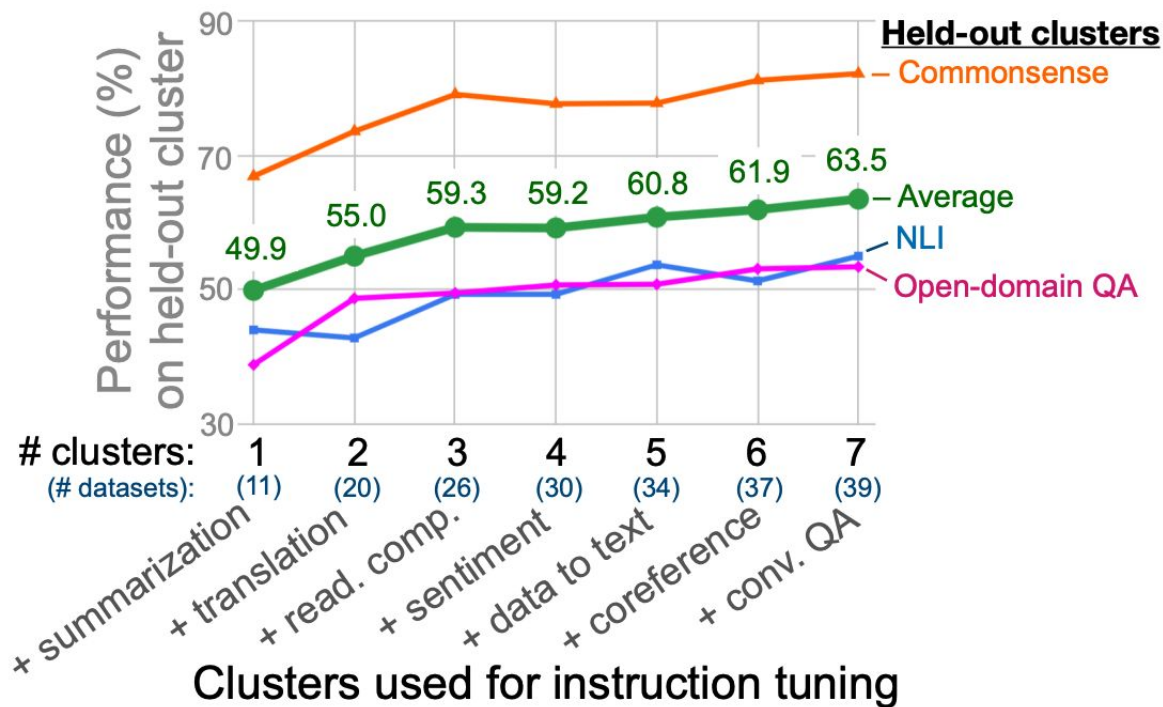


What Is Instruction Fine-Tuning?

Instruction Fine-Tuning (IFT) is a form of model training that enables models to better understand and act upon instructions. It is based on the idea that we can use everyday language to ask a model to perform a task and in return the model generates an accurate response in natural language.

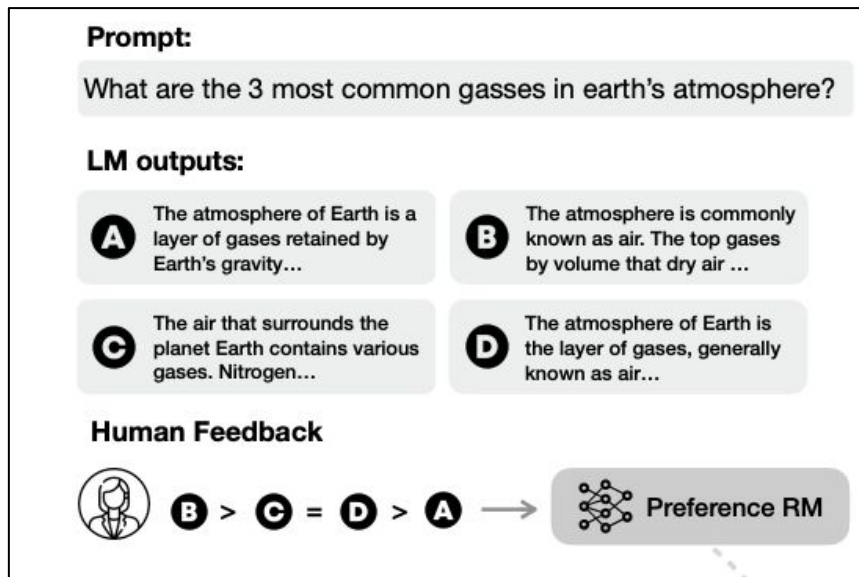


This combination – of multitask training and instruction style improves zero shot performance.



Finetuned language models are zero-shot learners (Wei et al., 2021).

Preference training aligns model behavior with human feedback by upweighting answers that humans find more meaningful in a third stage of optimization.




Gather feedback -> train model to align with feedback.

Upweight answers that humans find more meaningful.

We recently established new state-of-art in multilingual – by applying preference training on top of Aya 8b IFT.

Preference
Optimized (RLHF)
Aya 23 outperforms
llama 3, gemma,
mistral and
original Aya base.

Beats widely used
models across
languages covering
**half the world's
population!** 

RLHF Can Speak Many Languages: Unlocking Multilingual Preference Optimization for LLMs

John Dang
Cohere & Cohere For AI
johndang@cohere.com

Arash Ahmadian
Cohere & Cohere For AI
arash@cohere.com

Kelly Marchisio
Cohere
kelly@cohere.com

Julia Kreutzer
Cohere For AI
juliakreutzer@cohere.com

Ahmet Üstün
Cohere For AI
ahmet@cohere.com

Sara Hooker
Cohere For AI
sarahooker@cohere.com

Abstract

Preference optimization techniques have become a standard final stage for training state-of-art large language models (LLMs). However, despite widespread adoption, the vast majority of work to-date has focused on first-class citizen languages like English and Chinese. This captures a small fraction of the languages in the world, but also makes it unclear which aspects of current state-of-the-art research transfer to a multilingual setting. In this work, we perform an exhaustive study to achieve a new state-of-the-art in aligning multilingual LLMs. We introduce a novel, scalable method for generating high-quality multilingual feedback data to balance data coverage. We establish the benefits of cross-lingual transfer and increased dataset size in preference training. Our preference-trained model achieves a 54.4% win-rate against Aya 23 8B, the current state-of-the-art multilingual LLM in its parameter class, and a 69.5% win-rate or higher against widely used models like Gemma-1.1-7B-it, Llama-3-8B-Instruct, Mistral-7B-Instruct-v0.3. As a result of our study, we expand the frontier of alignment techniques to 23 languages covering half of the world's population.

Other algorithmic improvements which result in gains in performance without increasing training compute:

Other algorithmic improvements which result in gains in performance without increasing training compute:

1. Models enabled with tool use

2. Retrieval augmented models

3. Chain-of-thought

4. Best-of-n sampling

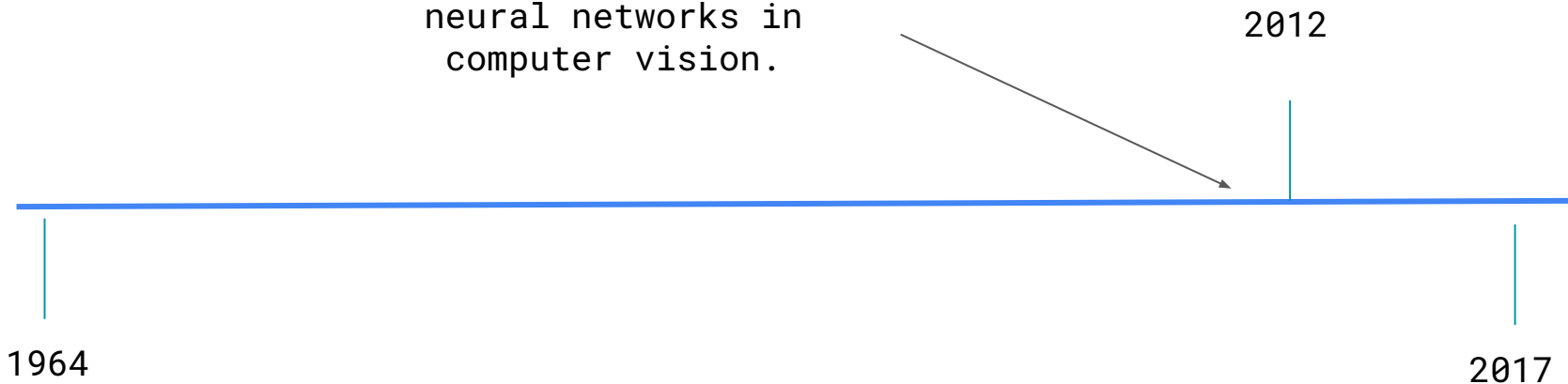
5. Distillation of synthetic data

6. Increasing context length

The role of architecture.

Overnight in 2012, everyone switched to deep neural networks.

Perseverance over decades led to the breakthrough of deep neural networks in computer vision.



2012: Convolutional Neural Networks

What aspects of the architecture improve efficiency?

ImageNet Classification with Deep Convolutional Neural Networks

Alex Krizhevsky
University of Toronto
kriz@cs.utoronto.ca

Ilya Sutskever
University of Toronto
ilya@cs.utoronto.ca

Geoffrey E. Hinton
University of Toronto
hinton@cs.utoronto.ca

Abstract

We trained a large, deep convolutional neural network to classify the 1.2 million high-resolution images in the ImageNet LSVRC-2010 contest into the 1000 different classes. On the test data, we achieved top-1 and top-5 error rates of 37.5% and 17.0% which is considerably better than the previous state-of-the-art. The neural network, which has 60 million parameters and 650,000 neurons, consists of five convolutional layers, some of which are followed by max-pooling layers, and three fully-connected layers with a final 1000-way softmax. To make training faster, we used non-saturating neurons and a very efficient GPU implementation of the convolution operation. To reduce overfitting in the fully-connected layers we employed a recently-developed regularization method called "dropout" that proved to be very effective. We also entered a variant of this model in the ILSVRC-2012 competition and achieved a winning top-5 test error rate of 15.3%, compared to 26.2% achieved by the second-best entry.

Efficiency gains:

Shared weights

shared weight kernels/filters that slide across the input image. convolutions to capture local patterns and features. This sharing of weights reduces the number of parameters in the network

Pooling Operations

Max-pooling and average pooling operations reduce the spatial dimensions of the feature maps, drastically reducing feature space.

Hierarchical Feature Extraction:

Each subsequent layer builds upon the features learned by previous layers.

2017: The Transformer, the culmination of a rich history of language modelling

What aspects of the architecture improve efficiency?

Attention Is All You Need

Ashish Vaswani*
Google Brain
avaswani@google.com

Noam Shazeer*
Google Brain
noam@google.com

Niki Parmar*
Google Research
nikip@google.com

Jakob Uszkoreit*
Google Research
usz@google.com

Llion Jones*
Google Research
llion@google.com

Aidan N. Gomez* †
University of Toronto
aidan@cs.toronto.edu

Łukasz Kaiser*
Google Brain
lukaszkaizer@google.com

Illia Polosukhin* ‡
illia.polosukhin@gmail.com

Efficiency gains:

Self-Attention Mechanism:

Self-attention layers ensure that the model can handle variable-length sequences, making it well-suited for language data.

Parallelizable Nature:

Process all the tokens in the input sequence in parallel. Parallel processing capability allows for faster training and inference compared to recursive or sequential models, which process tokens sequentially.

Lack of Recursion:

Transformers utilize position embeddings to encode the relative positions of tokens in the input sequence. This approach allows the model to capture the order of words without relying on recursive connections.

Architecture + algorithmic innovation
determine rate of return for compute.
So compute alone does not tell the
story of our field.

What is often missed in this statement is that our architectures also represent the ceiling in what is achievable through scaling.

Point of comparison: our Brain is incredibly energy efficient.

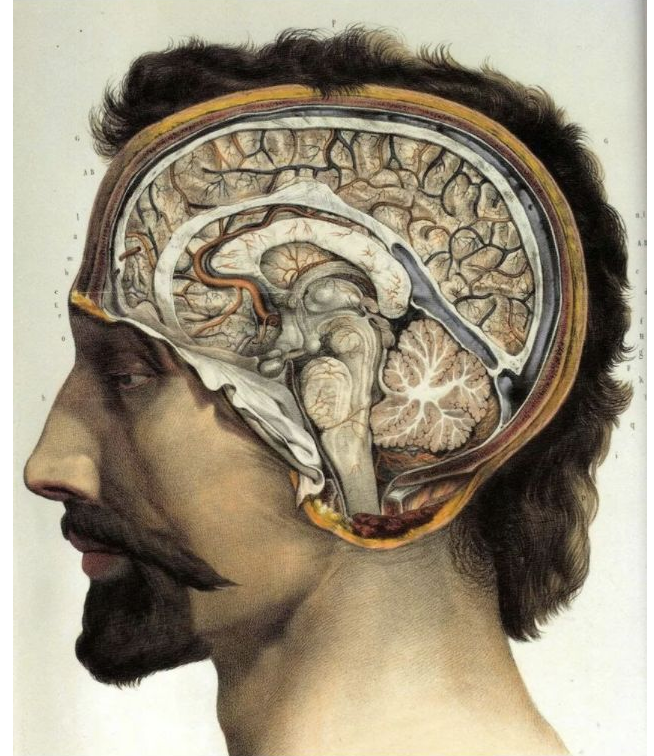
Has over 85 billion neurons but runs on the energy equivalent of an electric shaver

Key design choices to embed efficiency:

Specialized pathways

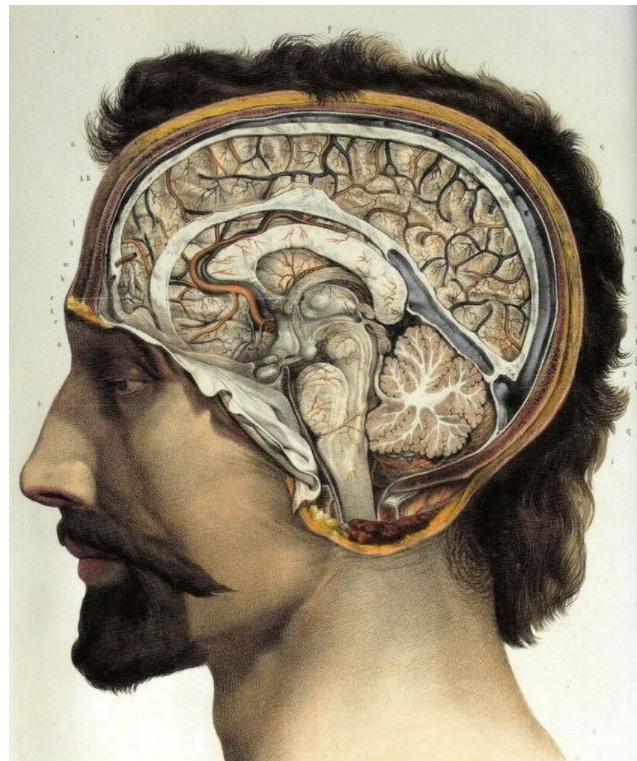
Simulate much of what we “see”

Log scale vision



Some aspects of what we do with deep neural networks is painfully inefficient.

- We do not have adaptive compute. Typically we see all examples same amount of time during training.
- Global updates mean all prior information is erased.
- Empirical risk minimization means while we optimize for average performance, it takes considerable more compute to model rare or infrequent artefacts.



Empirical risk minimization means we optimize to reduce average error:

This means it takes more capacity or longer training to learn rare features.

Majority of features are learnt early in training. Despite this most of training focuses on long-tail.

Majority of features can be learnt using small models. Scaling of size primarily benefits small tiny part of distribution.

Work with colleagues over last few years has focused on understanding what is lost and gained as we vary model size.

What Do Compressed Deep Neural Networks Forget?

Sara Hooker *
Google Brain

Aaron Courville
MILA

Gregory Clark
Google

Yann Dauphin
Google Brain

Andrea Frome
Google Brain

Abstract

Deep neural network pruning and quantization techniques have demonstrated it is possible to achieve high levels of compression with surprisingly little degradation to test set accuracy. However, this measure of performance conceals significant differences in how different classes and images are impacted by model compression techniques. We find that models with radically different numbers of weights have comparable top-line performance metrics but diverge considerably in behavior on a narrow subset of the dataset. This small subset of data points, which we term Pruning Identified Exemplars (PIEs) are systematically more impacted by the introduction of sparsity. Compression disproportionately impacts model performance on the underrepresented long tail of the data distribution. PIEs over-index on atypical or noisy images that are far more challenging for both humans and algorithms to classify. Our work provides intuition into the role of capacity in deep neural networks and the trade-offs incurred by compression. An understanding of this disparate impact is critical given the widespread deployment of compressed models in the wild.

CHARACTERISING BIAS IN COMPRESSED MODELS

Sara Hooker *
Google Research
shooker@google.com

Nyalleng Moorosi *
Google Research
nyalleng@google.com

Gregory Clark
Google
gregoryclark@google.com

Samy Bengio
Google Research
bengio@google.com

Emily Denton
Google Research
denton@google.com

ABSTRACT

The popularity and widespread use of pruning and quantization is driven by the severe resource constraints of deploying deep neural networks to environments with strict latency, memory and energy requirements. These techniques achieve high levels of compression with negligible impact on top-line metrics (top-1 and top-5 accuracy). However, overall accuracy hides disproportionately high errors on a small subset of examples; we call this subset Compression Identified Exemplars (CIE). We further establish that for CIE examples, compression amplifies existing algorithmic bias. Pruning disproportionately impacts performance on underrepresented features, which often coincides with considerations of fairness. Given that CIE is a relatively small subset but a great contributor of error in the model, we propose its use as a human-in-the-loop auditing tool to surface a tractable subset of the dataset for further inspection or annotation by a domain expert. We provide qualitative and quantitative support that CIE surfaces the most challenging examples in the data distribution for human-in-the-loop auditing.

The Low-Resource Double Bind: An Empirical Study of Pruning for Low-Resource Machine Translation

Orevaoghene Ahia
Masakhane NLP
oreva.ahia@gmail.com

Julia Kreutzer
Google Research
Masakhane NLP
jkreutzer@google.com

Sara Hooker
Google Research, Brain
shooker@google.com

Abstract

A “bigger is better” explosion in the number of parameters in deep neural networks has made it increasingly challenging to make state-of-the-art networks accessible in compute-restricted environments. Compression techniques have taken on renewed importance as a way to bridge the gap. However, evaluation of the trade-offs incurred by popular compression techniques has been centered on high-resource datasets. In this work, we instead consider the impact of compression in a data-limited regime. We introduce the term *low-resource double bind* to refer to the co-occurrence of data limitations and compute resource constraints. This is a common setting for NLP for low-resource languages, yet the trade-offs in

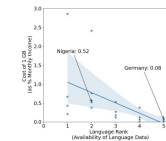


Figure 1: Cost of mobile data by country per language rank according to the taxonomy by Joshi et al. (2020).

Integrating Properties of Compression on Multilingual Models

Kelcehi Ogueji
University of Waterloo
kjoquej@uwaterloo.ca

Orevaoghene Ahia
University of Washington
oahia@cs.washington.edu

Ghemliké Onihale
Cohere For AI Community
lokeoni1@outlook.com

Sebastian Gehrmann
Google Research
sehrnann@google.com

Sara Hooker
Cohere For AI
sarahooker@cohere.com

Julia Kreutzer
Google Research
jkreutzer@google.com

Abstract

Multilingual models are often particularly dependent on scaling to generalize to a growing number of languages. Compression techniques are widely relied upon to reconcile the growth in model size with real-world resource constraints, but compression can have a disparate effect on model performance for low-resource languages. It is thus crucial to understand the trade-offs between scale, multilingualism, and compression. In this work, we propose an experimental framework to characterize the impact of sparsifying multilingual pre-trained language models during fine-tuning. Applying this framework to mBART named entity recognition models across 40 languages, we find that compression confers several intriguing and previously unseen generalization properties. In contrast to prior findings, we find that compression may improve model robustness over three models. We additionally observe that under certain specification regimes compression may aid, rather than disproportionately impact the performance of low-resource languages.

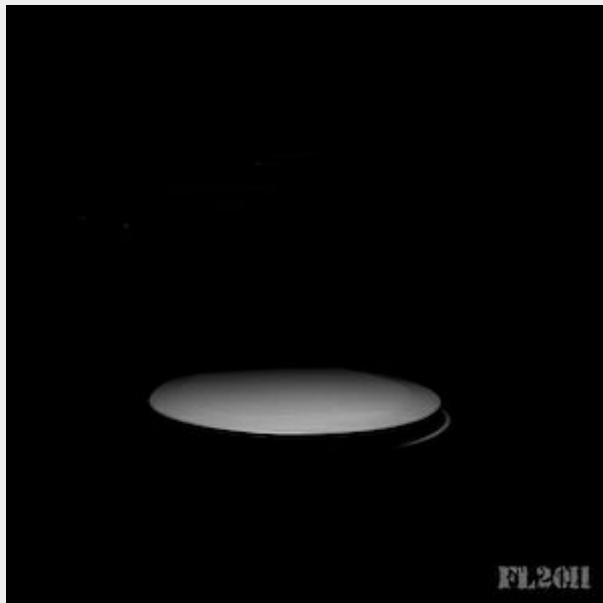
while maintaining comparable aggregate performance are widely used, such as quantization (Shen et al., 2020), compression (Michel et al., 2019; Lagunas et al., 2021) and distillation (Tsai et al., 2019; Siah et al., 2019; Pe et al., 2021). While most compression techniques have minimal impact on aggregate performance numbers (Gale et al., 2019; Li et al., 2020; How et al., 2020; Chen et al., 2021; Bai et al., 2020; ab Teodoro et al., 2021), the impact on individual sub-populations in the data, such as low-resource languages, can be far more severe (Hooker et al., 2019; Hooker et al., 2020; Ahia et al., 2021). Disparities in resource availability become more apparent at larger scale, both in terms of data and deployment resource availability. This makes compression all the more necessary, but also motivates a thorough consideration of the subsequent impact of compression on generalization. In this work, we develop an experimental framework to investigate the impact of compression during fine-tuning of pre-trained multilingual models which we apply to Named Entity Recognition (NER) across 40 languages of the WikiAnchors

[[Hooker et al. 2019, Hooker, Moorosi et al, 2020, Ahia et al. 2021, Ogueji et al. 2022, Marchisio 2024]]

Pruning Identified Exemplars (PIEs)

Data points where predictive behavior diverges between a population of independently trained compressed and non-compressed models.



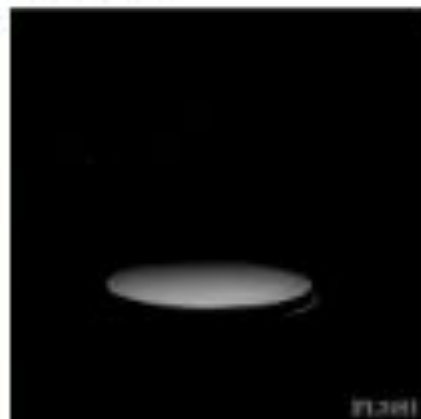


**ImageNet test-set.
True label?**

toilet seat



Non-PIE



PIE



**ImageNet test-set.
True label?**

espresso



Non-PIE



PIE



9

ImageNet test-set.
True label?

maze



Non-PIE



PIE



**ImageNet test-set.
True label?**

matchstick



Non-PIE



PIE

Noisy Data Points

- Data is improperly structured which corrupts information
 - Mislabeled
 - Severely corrupted
 - Multi-object

Misuse of parameters to represent these data points.

“Bad memorization”

Atypical Data Points or Challenging Exemplars

- Underrepresented vantage points (the long-tail of the dataset)
- Image classification entails fine grained task

Valuable use of parameters to represent these data points.

“Good memorization”

Noisy PIEs improperly structured multi-object images for single-image classification.



True Label:
desktop computer

Non-Pruned:
screen

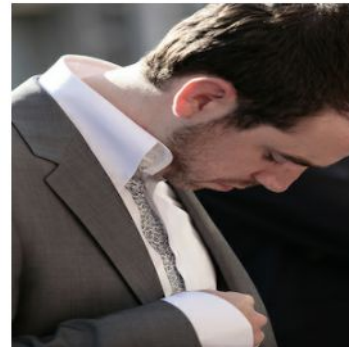
Pruned: monitor



True Label:
wine bottle

Non-Pruned:
red wine

Pruned: wine
bottle



True Label:
groom

Non-Pruned:
groom

Pruned: suit

Noisy PIEs Corrupted or incorrectly labeled data.



True Label:
restaurant

Non-Pruned:
meat loaf

Pruned: guacamole



True Label:
envelope

Non-Pruned:
dumbbell

Pruned: maraca



True Label:
tub

Non-Pruned:
cauldron

Pruned: wok

Atypical PIEs unusual vantage point or rare example



True Label:

toilet seat

Non-Pruned:

toilet seat

Pruned: folding

chair



True Label:

bathtub

Non-Pruned:

bathtub

Pruned: cucumbe

r



True Label:

plastic bag

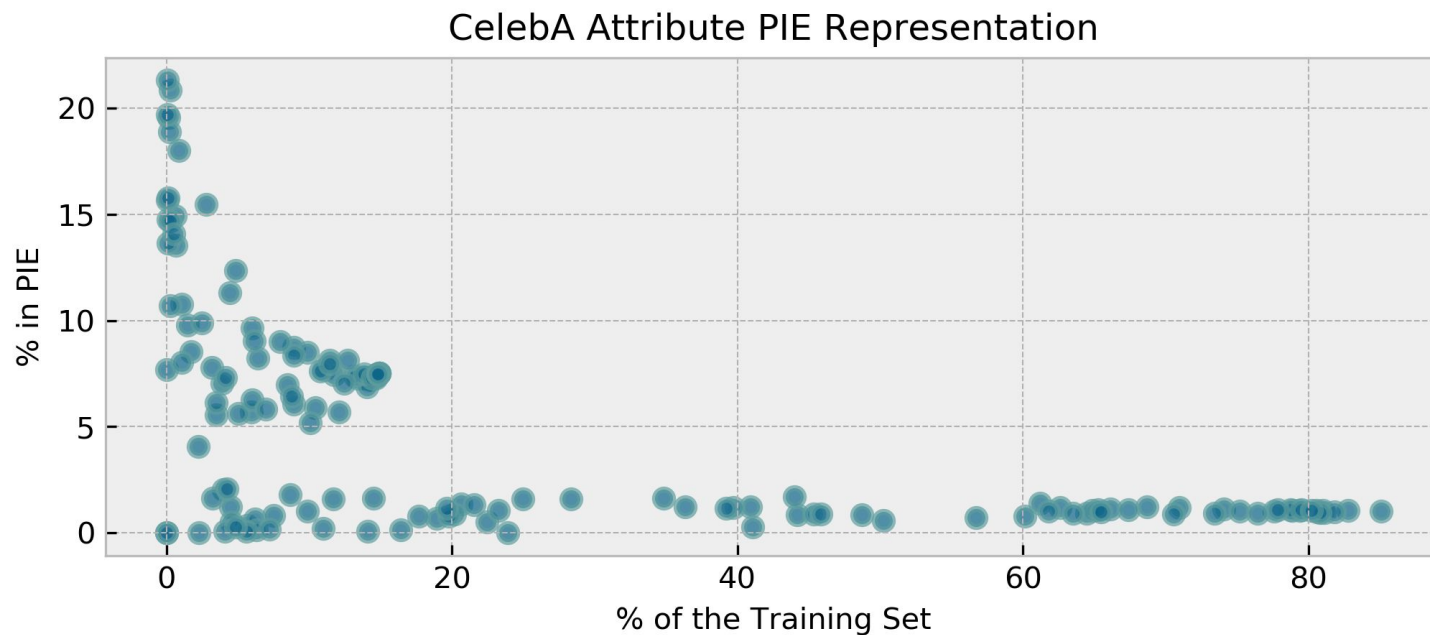
Non-Pruned:

gown

Pruned: plastic

bag

PIEs over-index on the long-tail of underrepresented attributes.



Attribute Proportion of CelebA Training Data vs. relative representation in PIE

It is worth emphasizing this finding: We lose the long-tail when we remove the majority of all training weights.

Put differently, we are using the majority of our weights to encode a useful representation for a small fraction of our training distribution.



0 %

90 %

Overparameterized
Dense Model

Model with 90%
weights removed

When we scale models, we are paying an enormous cost to learn a small slice of the distribution.



So, where do we go from here.

How should we make better use of capacity? How do we avoid the cost of representing the long-tail?

Directions that we have been working on:

1. Data pruning or Weighting

“Spending more capacity on the data points we care about”

2. Specialized/adaptive compute


“Routing capacity to avoid applying all weights to every example”

Data pruning.

Noisy Data Points

- Data is improperly structured which corrupts information
 - Mislabeled
 - Severely corrupted
 - Multi-object

Can we identify and remove these noisy data points?



- How do we do this at scale?

Misuse of parameters to represent these data points.

“Bad memorization”

Much of our recent work over the last year has focused on data pruning, prioritization of examples.

When Less is More: Investigating Data Pruning for Pretraining LLMs at Scale

Max Marion
Cohere for AI
maxwell@cohere.com

Ahmet Üstün
Cohere for AI
ahmet@cohere.com

Luiza Pozzobon
Cohere for AI
luiza@cohere.com

Alex Wang
Cohere
alexwang@cohere.com

Marzieh Fadaee
Cohere for AI
marzieh@cohere.com

Sara Hooker
Cohere for AI
sarahooker@cohere.com

Abstract

Large volumes of text data have contributed significantly to the development of large language models (LLMs) in recent years. This data is typically acquired by scraping the internet, leading to pretraining datasets comprised of noisy web text. To date, efforts to prune these datasets down to a higher quality subset have relied on hand-crafted heuristics encoded as rule-based filters. In this work, we take a wider view and explore scalable estimates of data quality that can be used to systematically measure the quality of pretraining data. We perform a rigorous comparison at scale of the simple data quality estimator of perplexity, as well as more sophisticated and computationally intensive estimates of the Error L2-Norm and memorization. These metrics are used to rank and prune pretraining corpora, and we subsequently compare LLMs trained on these pruned datasets. Surprisingly, we find that the simple technique of perplexity outperforms our more computationally expensive scoring methods. We improve over our no-pruning baseline while training on as little as 30% of the original training dataset. Our work sets the foundation for unexplored strategies in automatically curating high quality corpora and suggests the majority of pretraining data can be removed while retaining performance.

Pretraining Scale

[[[Marion et al. 2023](#)]]

Aya Dataset: An Open-Access Collection for Multilingual Instruction Tuning

Shivalika Singh^{✳1}, Freddie Vargus^{✳1}, Daniel D'souza^{✳1}, Börje F. Karlsson^{✳2},
Abinaya Mahendiran^{✳1}, Wei-Yin Ko^{✳3}, Herumb Shandilya^{✳1}, Jay Patel⁴,
Deividas Maticunas¹, Laura O'Mahony⁵, Mike Zhang⁶, Ramith Hettiarachchi⁷,
Joseph Wilson⁸, Marina Machado³, Luisa Souza Moura³, Dominik Krzemiński¹,
Hakimeh Fadaei¹, Irem Ergün³, Ifeoma Okoh¹, Aisha Alaagib¹,
Oshan Mudannayake¹, Zaid Alyafeai⁹, Vu Minh Chien¹, Sebastian Ruder³,
Surya Guthikonda¹, Emad A. Alghamdi¹⁰, Sebastian Gehrmann¹¹,
Niklas Muennighoff¹, Max Bartolo³, Julia Kreutzer¹², Ahmet Üstün¹²,
Marzieh Fadaee¹², and Sara Hooker¹²

¹Cohere For AI Community, ²Beijing Academy of Artificial Intelligence, ³Cohere, ⁴Binghamton University,
⁵University of Limerick, ⁶IT University of Copenhagen, ⁷MIT, ⁸University of Toronto, ⁹King Fahd University of
Petroleum and Minerals, ¹⁰King Abdulaziz University, ASAS.AI, ¹¹Bloomberg LP, ¹²Cohere For AI

Corresponding authors: Shivalika Singh <shivalikasingh95@gmail.com>, Marzieh Fadaee <marzieh@cohere.com>,
Sara Hooker <sarahooker@cohere.com>

Instruction Finetuning Pruning and Dataset Weighting

[[[Singh et al. 2023](#)]]

Which Prompts Make The Difference? Data Prioritization For Efficient Human LLM Evaluation

Meriem Boudir
Cohere for AI
meri.boudir@gmail.com

Edward Kim
Cohere
edward@cohere.com

Beyza Ermiş
Cohere for AI
beyza@cohere.com

Marzieh Fadaee
Cohere for AI
marzieh@cohere.com

Sara Hooker
Cohere for AI
sarahooker@cohere.com

Abstract

Human evaluation is increasingly critical for assessing large language models, capturing linguistic nuances, and reflecting user preferences more accurately than traditional automated metrics. However, the resource-intensive nature of this type of annotation process poses significant challenges. The key question driving our work: *is it feasible to minimize human-in-the-loop feedback by prioritizing data instances which most effectively distinguish between models?* We evaluate several metric-based methods and find that these metrics enhance the efficiency of human evaluations by minimizing the number of required annotations, thus saving time and cost, while ensuring a robust performance evaluation. We show that our method is effective across widely used model families, reducing instances of indecisive (or “tie”) outcomes by up to 54% compared to a random sample when focusing on the top-20 percentile of prioritized instances. This potential reduction in required human effort positions our approach as a valuable strategy in future large language model evaluations.

Prioritizing human annotation

[[[Boudir et al. 2023](#)]]

Removing, reprioritizing, generating synthetic data has become a crucial pipeline step of achieving state of art performance.

Critical Learning Periods: Leveraging Early Training Dynamics for Efficient Data Pruning

Everlyn Asiko Chimoto^{1,2,3} Jay Gala^{1,5} Orevaoghene Abia^{1,6}
Julia Kreutzer⁷ Bruce A. Bassett^{2,4} Sara Hooker⁷

¹Cohere For AI Community ²University of Cape Town, South Africa

³African Institute for Mathematical Sciences ⁴South African Astronomical Observatory

⁵Mohamed bin Zayed University of Artificial Intelligence

⁶University of Washington ⁷Cohere For AI

Abstract

Neural Machine Translation models are extremely data and compute-hungry. However, not all data points contribute equally to model training and generalization. Data pruning to remove the low-value data points has the benefit of drastically reducing the compute budget without a significant drop in model performance. In this paper, we propose a new data pruning technique: *Checkpoints Across Time (CAT)*, that leverages early model training dynamics to identify the most relevant data points for model performance. We benchmark *CAT* against several data pruning techniques including COMET-QE, LASER and LaBSE. We find that *CAT* outperforms the benchmarks on Indo-European languages on multiple test sets. When applied to English-German, English-French and English-Swahili translation tasks, *CAT* achieves comparable performance to using the full dataset, while pruning up to 50% of training data. We inspect the data points that *CAT* selects

Aya 23: Open Weight Releases to Further Multilingual Progress

Viraat Aryabumi^{*1}, John Dang¹, Dwarak Talupuru²,
Saurabh Dash¹, David Cairuz², Hangyu Lin², Bharat Venkitesh²,
Madeline Smith¹, Jon Ander Campos², Yi Chern Tan²,
Kelly Marchisio², Max Bartolo², Sebastian Ruder², Acyr Locatelli²,
Julia Kreutzer¹, Nick Frosst², Aidan Gomez², Phil Blunsom²,
Marzieh Fadaee¹, Ahmet Üstün^{*1}, and Sara Hooker^{*1}

¹Cohere For AI, ²Cohere

Corresponding authors: Viraat Aryabumi <viraat@cohere.com>, Ahmet Üstün <ahmet@cohere.com>, Sara Hooker <sarahooker@cohere.com>

Abstract

This technical report introduces *Aya 23*, a family of multilingual language models. *Aya 23* builds on the recent release of the *Aya* model [Üstün et al., 2024], focusing on pairing a highly performant pre-trained model with the recently released *Aya* collection [Singh et al., 2024]. The result is a powerful multilingual large language model serving 23 languages, **expanding state-of-art language modeling capabilities to approximately half of the world's population**. The *Aya* model covered 101 languages whereas *Aya 23* is an experiment in depth vs breadth, exploring the impact of allocating more capacity to fewer languages that are included during pre-training. *Aya 23* outperforms both previous massively multilingual models like *Aya 101* for the languages it covers, as well as widely used models like Gemma, Mistral and Mixtral on an extensive range of discriminative and generative tasks. We release the open weights for both the 8B and 35B models as part of our continued commitment for expanding access to multilingual progress.

RLHF Can Speak Many Languages: Unlocking Multilingual Preference Optimization for LLMs

John Dang
Cohere For AI
johndang@cohere.com

Arash Ahmadian
Cohere @ Cohere For AI
arash@cohere.com

Kelly Marchisio
Cohere
kelly@cohere.com

Julia Kreutzer
Cohere For AI
juliakreutzer@cohere.com

Ahmet Üstün
Cohere For AI
ahmet@cohere.com

Sara Hooker
Cohere For AI
sarahooker@cohere.com

Abstract

Preference optimization techniques have become a standard final stage for training state-of-art large language models (LLMs). However, despite widespread adoption, the vast majority of work to-date has focused on first-class citizen languages like English and Chinese. This captures a small fraction of the languages in the world, but also makes it unclear which aspects of current state-of-the-art research transfer to a multilingual setting. In this work, we perform an exhaustive study to achieve a new state-of-the-art in aligning multilingual LLMs. We introduce a novel, scalable method for generating high-quality multilingual feedback data to balance data coverage. We establish the benefits of cross-lingual transfer and increased dataset size in preference training. Our preference-trained model achieves a 54.4% win-rate against *Aya 23* 8B, the current state-of-the-art multilingual LLM in its parameter class, and a 69.5% win-rate or higher against widely used models like Gemma-1.1-7B-it, Llama-3-8B-Instruct, Mistral-7B-Instruct-v0.3. As a result of our study, we expand the frontier of alignment techniques to 23 languages covering half of the world's population.

Leveraging early
training signal

[[[Chimoto et al. 2023](#)]]

Data pruning +
synthetic data

[[[Aryadumi et al. 2024](#)]]

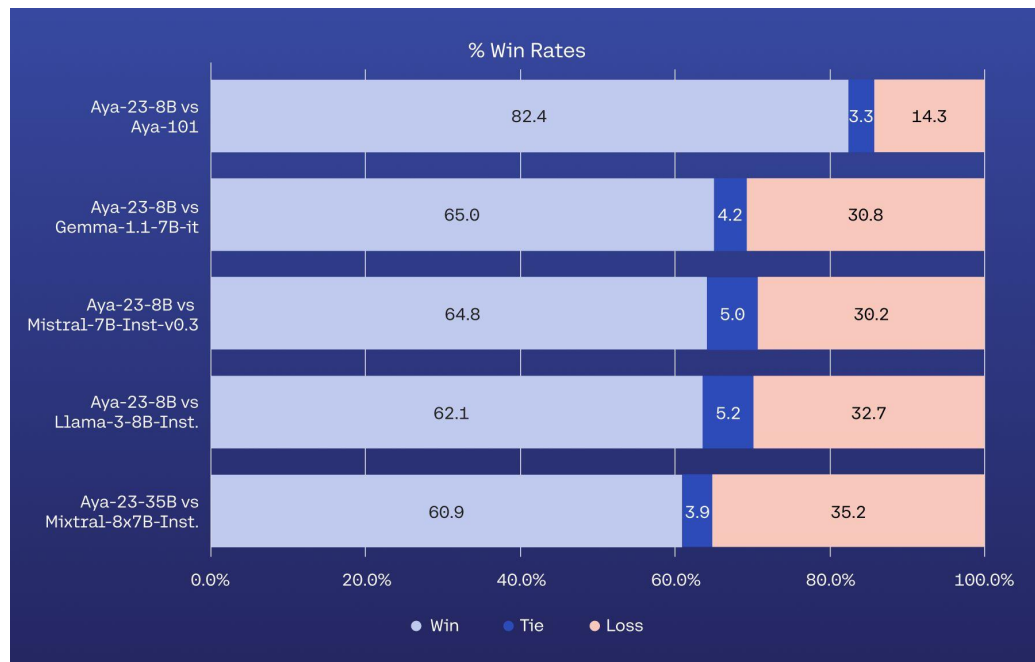
Preference training
synthetic data.

[Dang et al.
2024](#)

High quality synthetic data also reduces training time and improves performance.

We recently released Aya 8b – a best in class small multilingual model which outperforms Gemma, Llama 3 and Mistral.

Most of the gains came from distillation of synthetic data from larger more performant models.



We have also done work on “active inheritance,” moving optimization to the data space to guide model behavior towards non-differentiable objectives.

Only very recently feasible to do this – historically steering data collection far too expensive. Dataset treated as static.

✧ Cohere For AI

LLM See, LLM Do: Guiding Data Generation to Target Non-Differentiable Objectives

Lufsa Shimabucoro[†]
Cohere For AI

Sebastian Ruder
Cohere

Julia Kreutzer
Cohere For AI

Marzieh Fadaee[†]
Cohere For AI

Sara Hooker[†]
Cohere For AI

Abstract

The widespread adoption of synthetic data raises new questions about how models generating the data can influence other large language models (LLMs) via distilled data. To start, our work exhaustively characterizes the impact of *passive inheritance* of model properties by systematically studying the consequences of synthetic data integration. We provide one of the most comprehensive studies to-date of how the source of synthetic data shapes models’ internal biases, calibration and generations’ textual attributes and preferences. We find that models are surprisingly sensitive towards certain attributes even when the synthetic data prompts appear “neutral,” which invites the question whether this sensitivity can be exploited for good.

Our findings invite the question *can we explicitly steer the models towards the properties we want at test time by exploiting the data generation process?* This would have historically been considered infeasible due to the cost of collecting data with a specific characteristic or objective in mind. However, improvement in the quality of synthetic data, as well as a shift towards general-purpose models designed to follow a diverse way of instructions, means this question is timely. We propose *active inheritance* as a term to describe intentionally constraining synthetic data, according to a

[Shimabucoro et al. 2024](#)

Specialized
and/or
Adaptive
compute

2.

Specialized/adaptive compute



“Routing capacity to avoid
applying all weights to every
example”

Adaptive
compute –
spend more time
on the data
points we care
about.



Atypical Data Points or Challenging Exemplars

- Underrepresented vantage points (the long-tail of the dataset)
- Image classification entails fine grained task

Valuable use of parameters to
represent these data points.

“Good memorization”

Our recent work on Mixtures of Experts is focused on Specialized/Adaptive compute.

Specialized/adaptive compute

“Routing capacity to avoid applying all weights to every example”

Pushing Mixture of Experts to the Limit: Extremely Parameter Efficient MoE for Instruction Tuning

Ted Zadouri
Cohere for AI
ted@cohere.com

Ahmet Üstün
Cohere for AI
ahmet@cohere.com

Arash Ahmadian[†]
Cohere for AI
arash@cohere.com

Beyza Ermiş
Cohere For AI
beyza@cohere.com

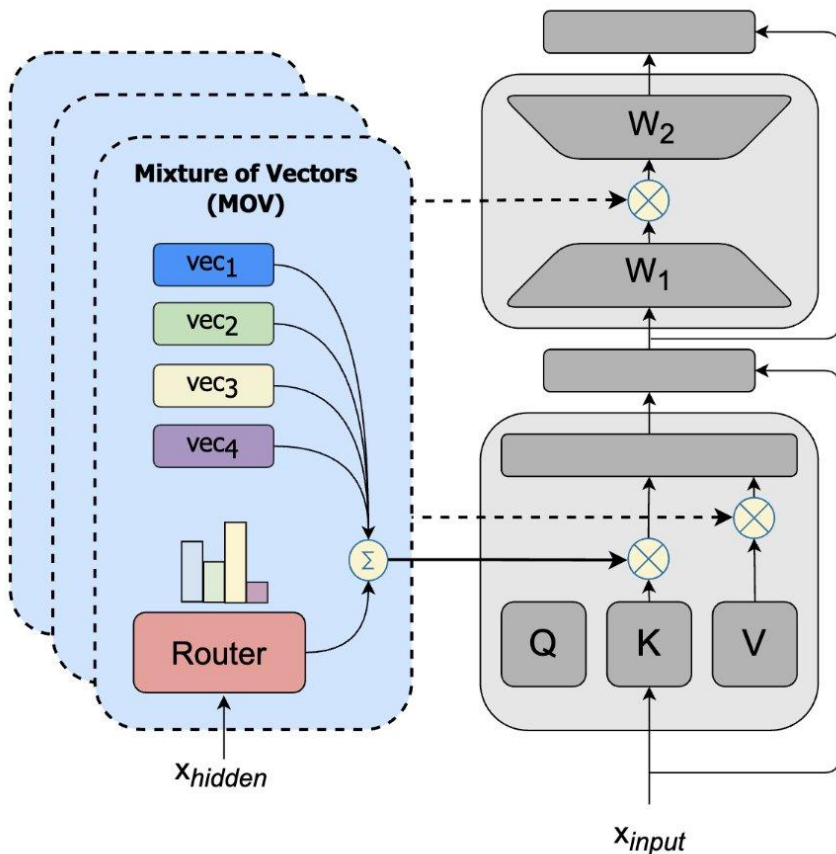
Acyr Locatelli
Cohere
acyr@cohere.com

Sara Hooker
Cohere for AI
sarahooker@cohere.com

Abstract

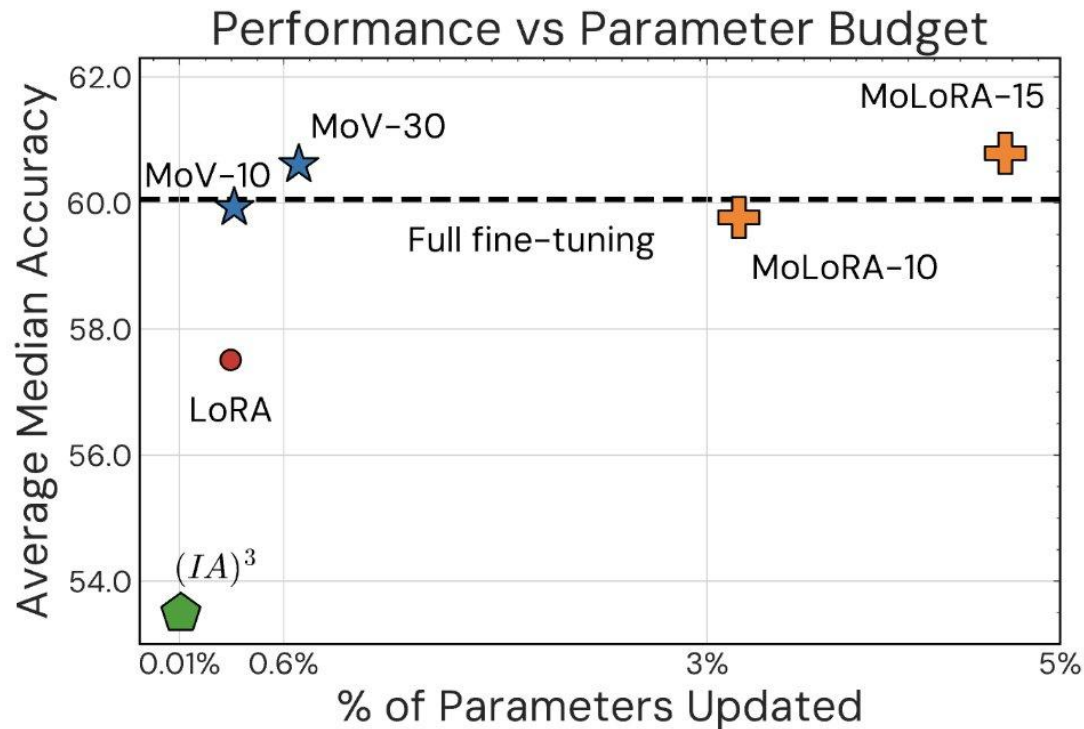
The Mixture of Experts (MoE) is a widely known neural architecture where an ensemble of specialized sub-models optimizes overall performance with a constant computational cost. However, conventional MoEs pose challenges at scale due to the need to store all experts in memory. In this paper, we push MoE to the limit. We propose extremely parameter-efficient MoE by uniquely combining MoE architecture with lightweight experts. Our MoE architecture outperforms standard parameter-efficient fine-tuning (PEFT) methods and is on par with full fine-tuning by only updating the lightweight experts – less than 1% of an 11B parameters model. Furthermore, our method generalizes to unseen tasks as it does not depend on any prior task knowledge. Our research underscores the versatility of the mixture of experts architecture, showcasing its ability to deliver robust performance even when subjected to rigorous parameter constraints. Our code used in all the experiments is publicly available here: <https://github.com/for-ai/parameter-efficient-moe>.

Full model fine-tuning? Costly for large model sizes



- MoEs style architecture with ultra-lightweight experts
- Mixture of IA3 vectors (MoV) or LoRA adapters (MoLoRA), as experts.
- During fine-tuning, only vectors/experts and routers are updated for each layer.

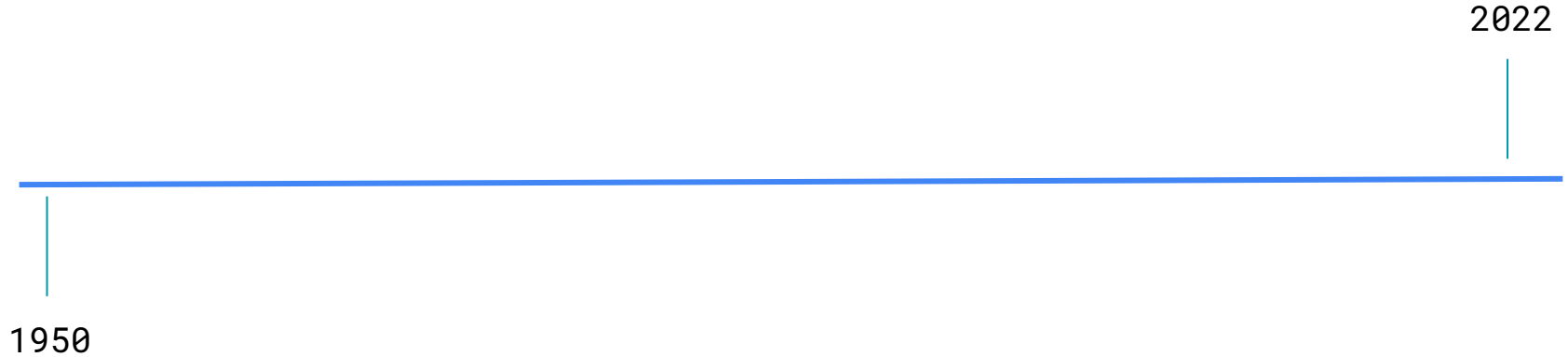
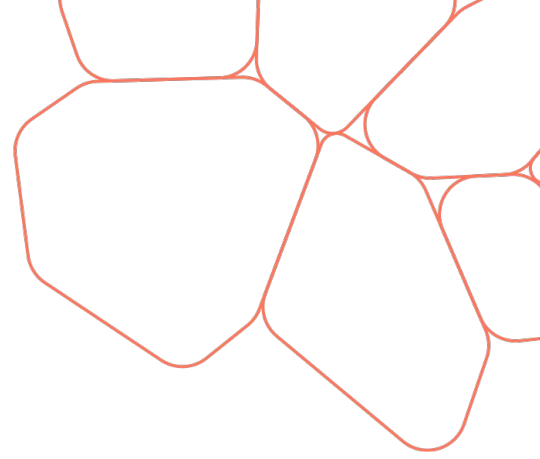
Performance of a Mixture of “Ultra-Lightweight” Experts



- Our method outperforms standard PEFT methods like IA3 or LoRA.
- Using a mixture of IA3 style vectors (MoV), we update just **0.68%** of the model parameters, boosting performance by **14.57%** over its counterpart IA3.

Parting thoughts.

Modern computer science as
a field has only existed
for the last 77 years.



It is very possible that the next breakthrough will require a fundamentally different way of modelling the world

with a different combination of hardware, software and algorithm.



It is the least interesting thing to throw compute at a problem.
Increasingly, we should justify additional complexity and bend
scaling curves by focusing on efficiency.



Key takeaways:

- We are spending a disproportionate amount of capacity learning the long tail.
- A more viable path forward is adaptive capacity – spending more time on the parts of the data distribution we want to learn (either data pruning, data selection – or in formulation of algorithms (moe))
- There is a ceiling to returns from compute – we are currently building a ladder to the moon.

Questions?

Intriguing Properties of Quantization at Scale Arash Ahmadian, Saurabh Dash, Hongyu Chen, Bharat Venkitesh, Stephen Gou, Phil Blunsom, Ahmet Üstün, Sara Hooker [[paper link](#)]

When Less is More: Investigating Data Pruning for Pretraining LLMs at Scale Max Marion, Ahmet Üstün, Luiza Pozzobon, Alex Wang, Marzieh Fadaee, Sara Hooker [[paper link](#)]

Pushing Mixture of Experts to the Limit: Extremely Parameter Efficient MoE for Instruction Tuning Ted Zadouri, Ahmet Üstün, Arash Ahmadian, Beyza Ermiş, Acyr Locatelli, Sara Hooker [[paper link](#)]

Which Prompts Make The Difference? Data Prioritization For Efficient Human LLM Evaluation Meriem Boubdir, Edward Kim, Beyza Ermiş, Marzieh Fadaee, Sara Hooker [[paper link](#)]

Feel free to reach out if any of these ideas is relevant to work you are doing..

Final takeaways:

Recent breakthroughs in NLP - combination of changes in optimization, scale (of both data and weights)

Key challenge - efficiency of our chosen representation. The relationship between weights and generalization is not well understood.

Promising directions of improving efficiency – includes both algorithmic, hardware-software and data space.

Tension between theoretical and practical motivations – some cherished theoretical techniques do not produce speed ups.

Email: sarahooker@cohere.com