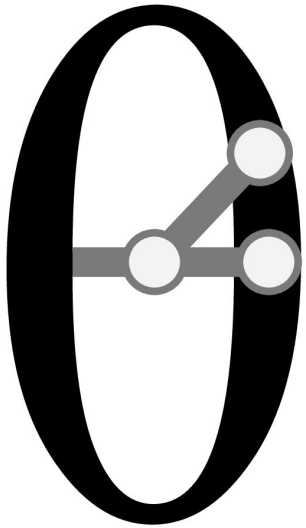
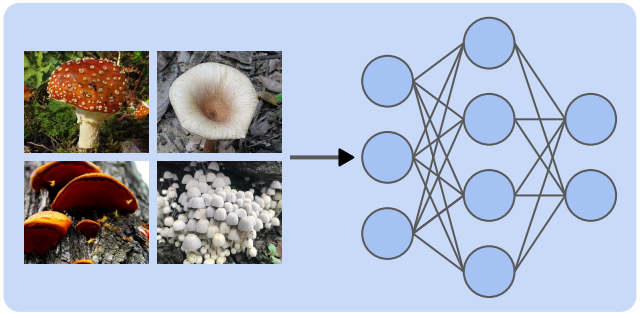
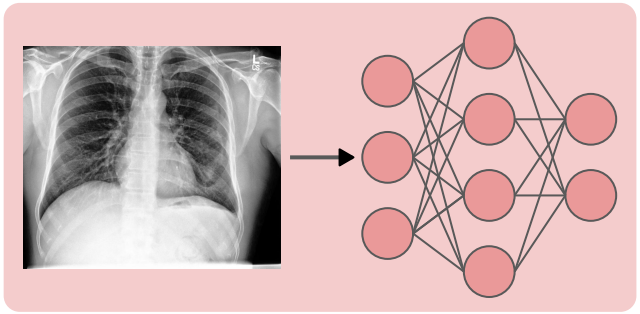
Building Machine Learning Models like Open-Source Software with `git-theta`

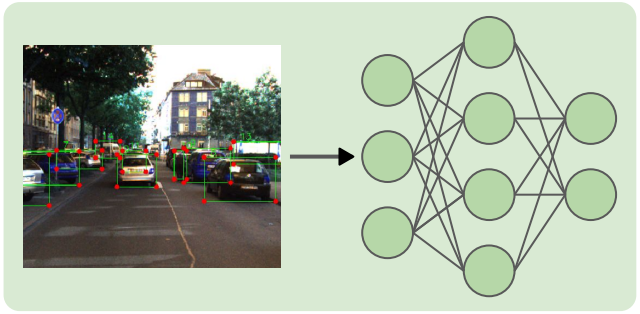


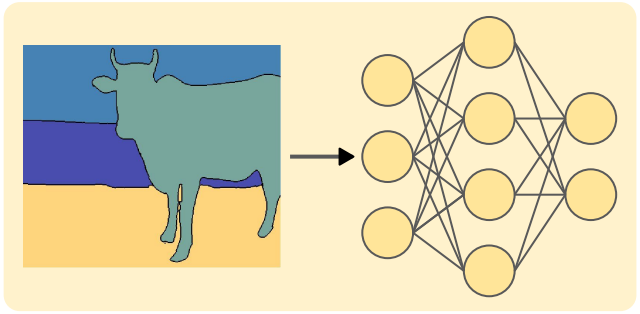
Nikhil Kandpal & Colin Raffel

Deep learning circa 2013 – training models from scratch

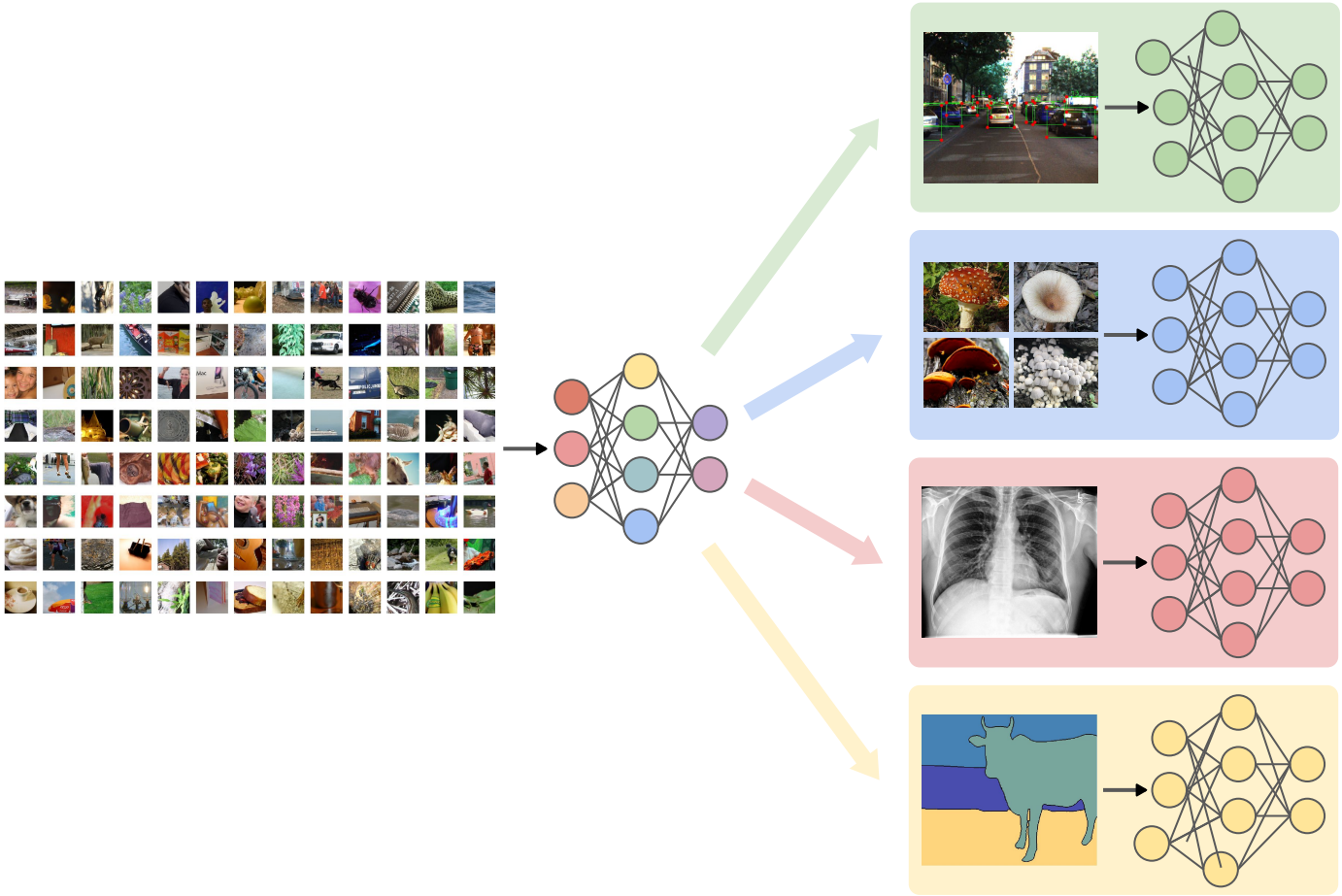




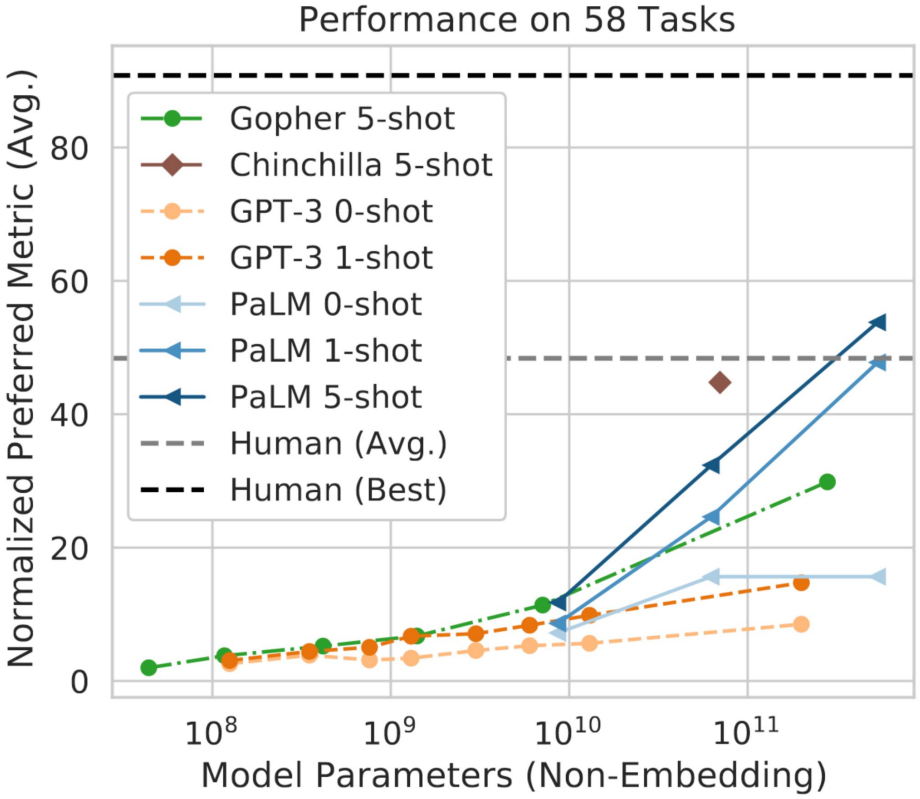


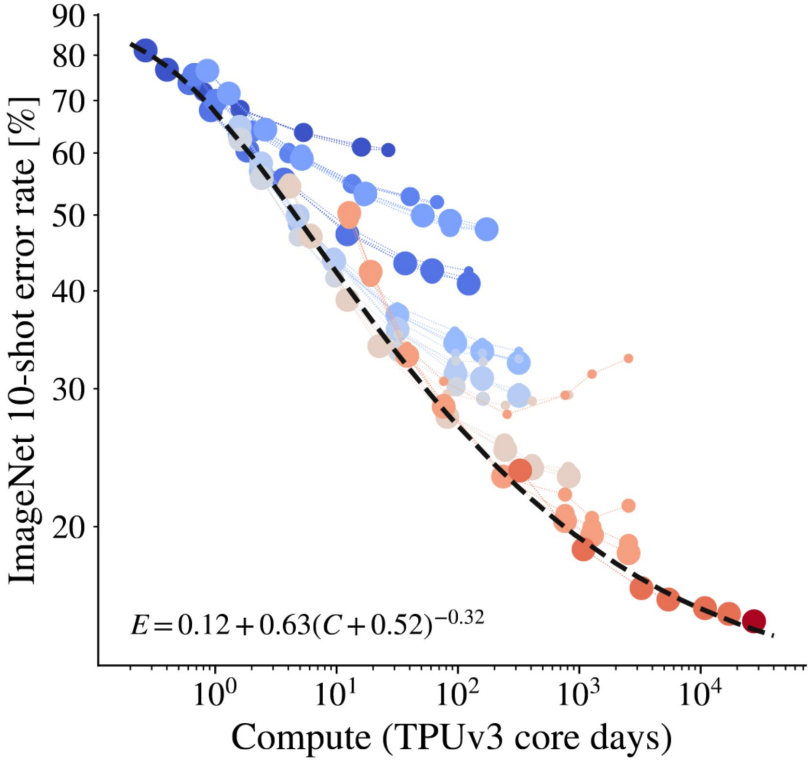


Deep learning in 2023 – pre-train then adapt



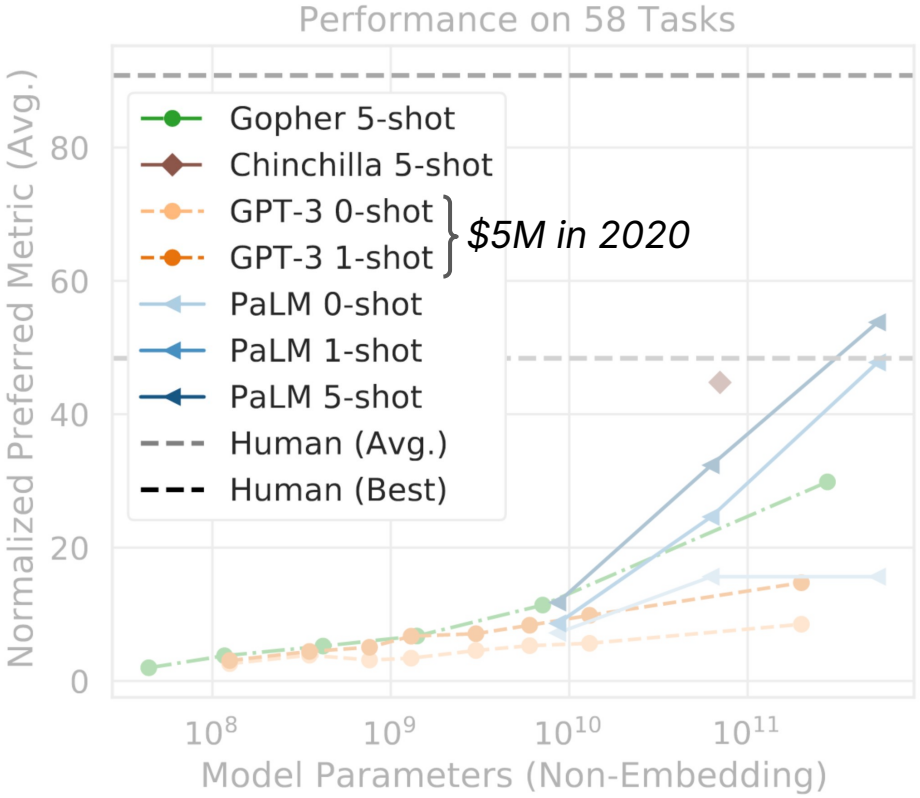
The benefits – and costs – of scale

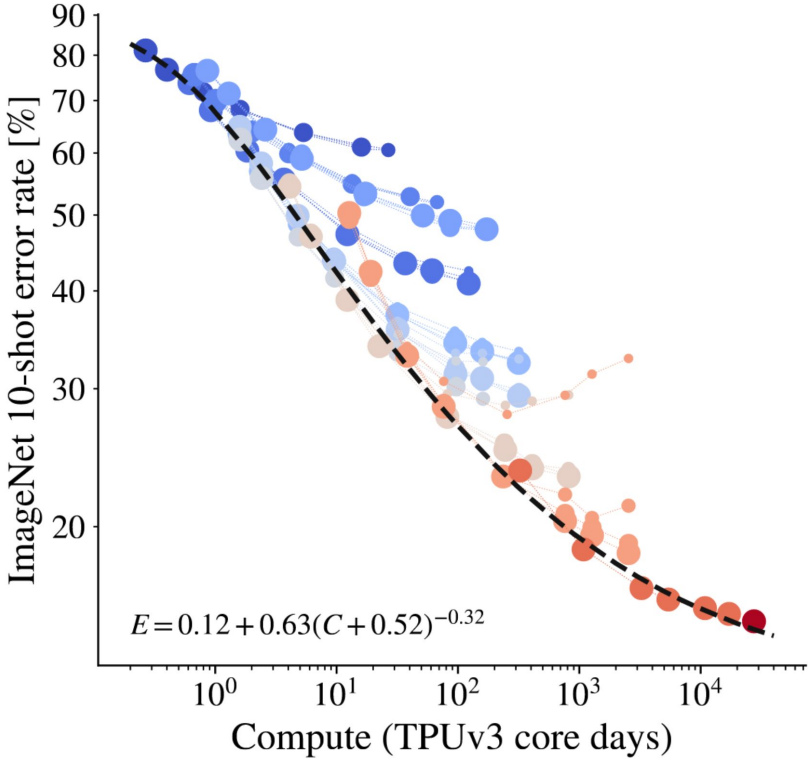




From "PaLM: Scaling Language Modeling with Pathways" by Chowdhery et al. and "Scaling Vision Transformers" by Zhai et al.

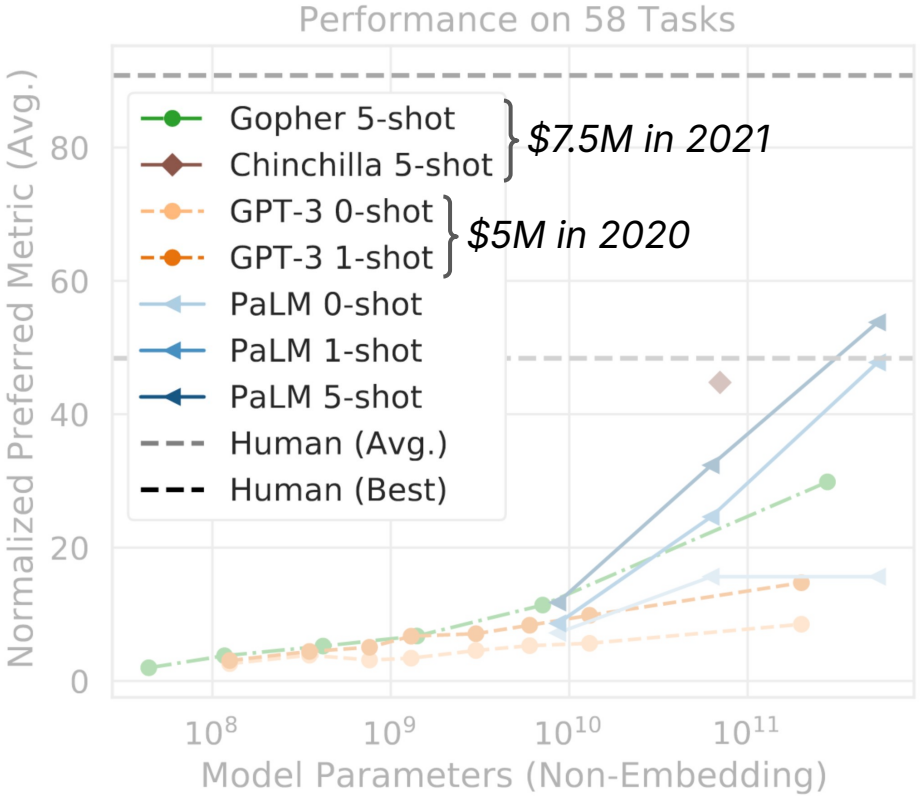
The benefits – and costs – of scale

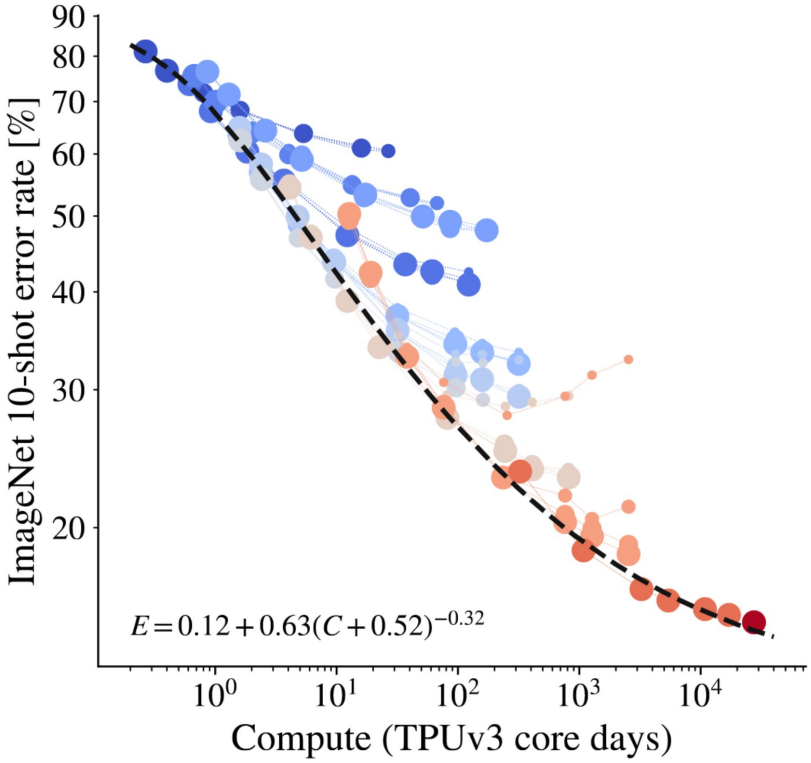




From "PaLM: Scaling Language Modeling with Pathways" by Chowdhery et al. and "Scaling Vision Transformers" by Zhai et al.

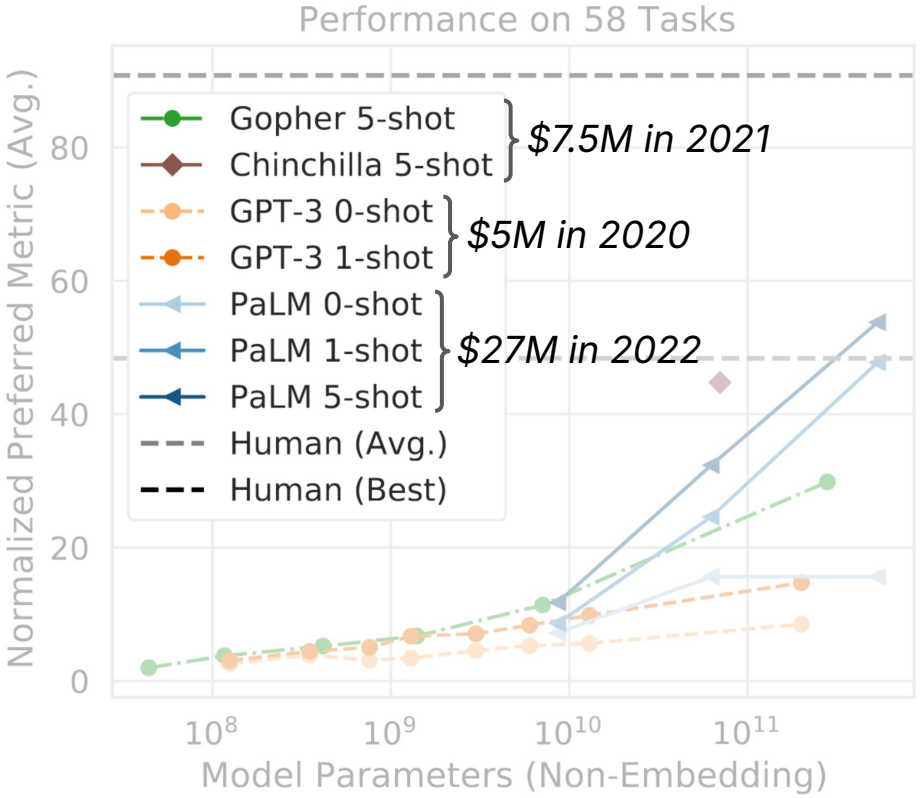
The benefits – and costs – of scale

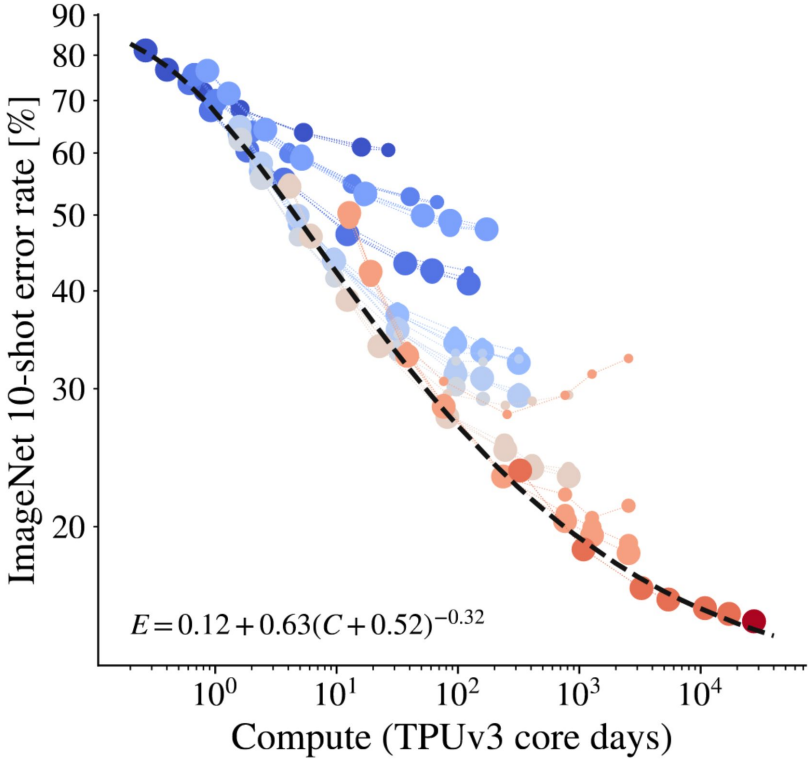




From "PaLM: Scaling Language Modeling with Pathways" by Chowdhery et al. and "Scaling Vision Transformers" by Zhai et al.

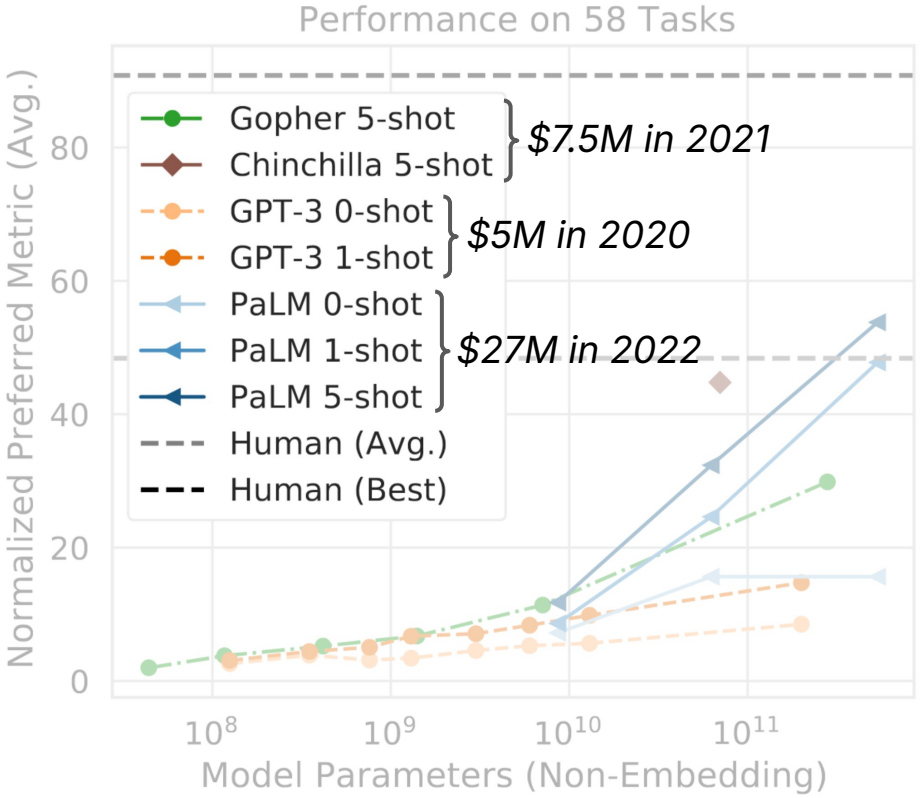
The benefits – and costs – of scale

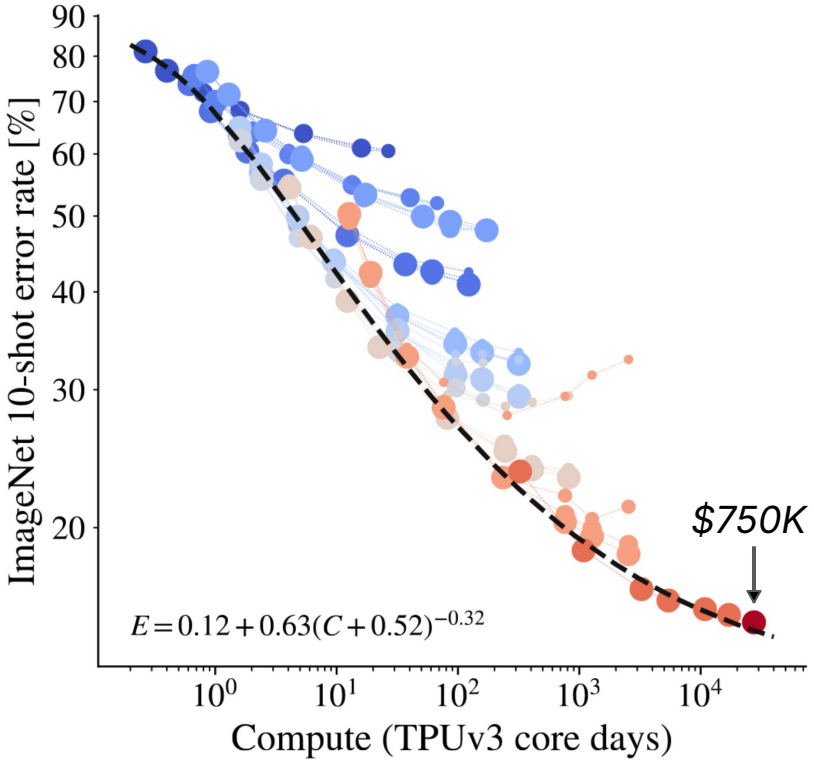




From "PaLM: Scaling Language Modeling with Pathways" by Chowdhery et al. and "Scaling Vision Transformers" by Zhai et al.

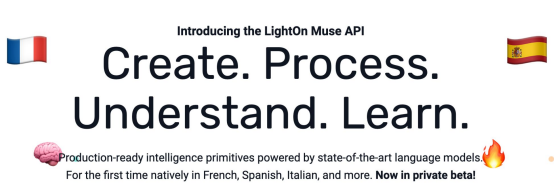
The benefits – and costs – of scale

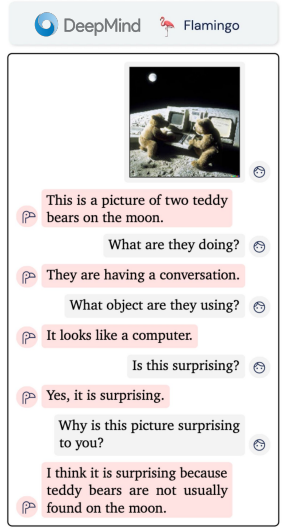


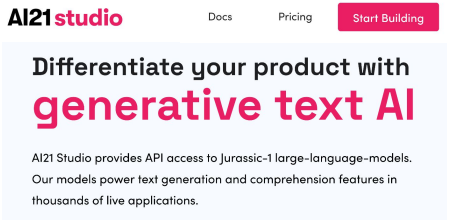


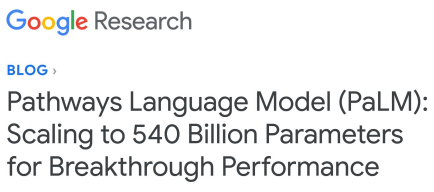
From "PaLM: Scaling Language Modeling with Pathways" by Chowdhery et al. and "Scaling Vision Transformers" by Zhai et al.

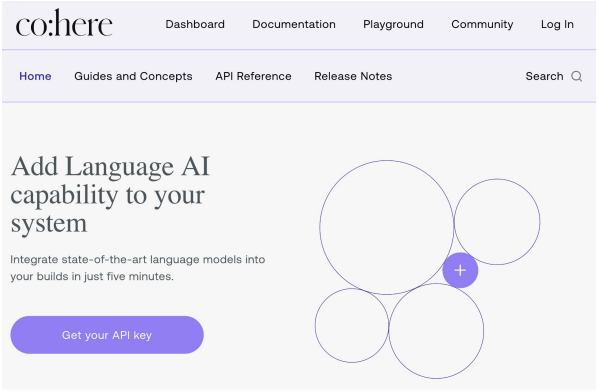
Increased costs have decreased sharing

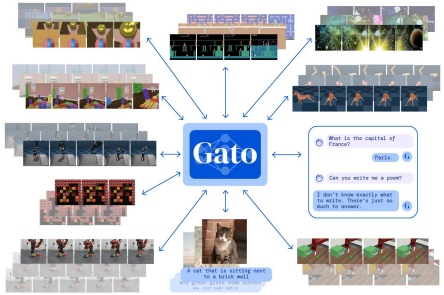








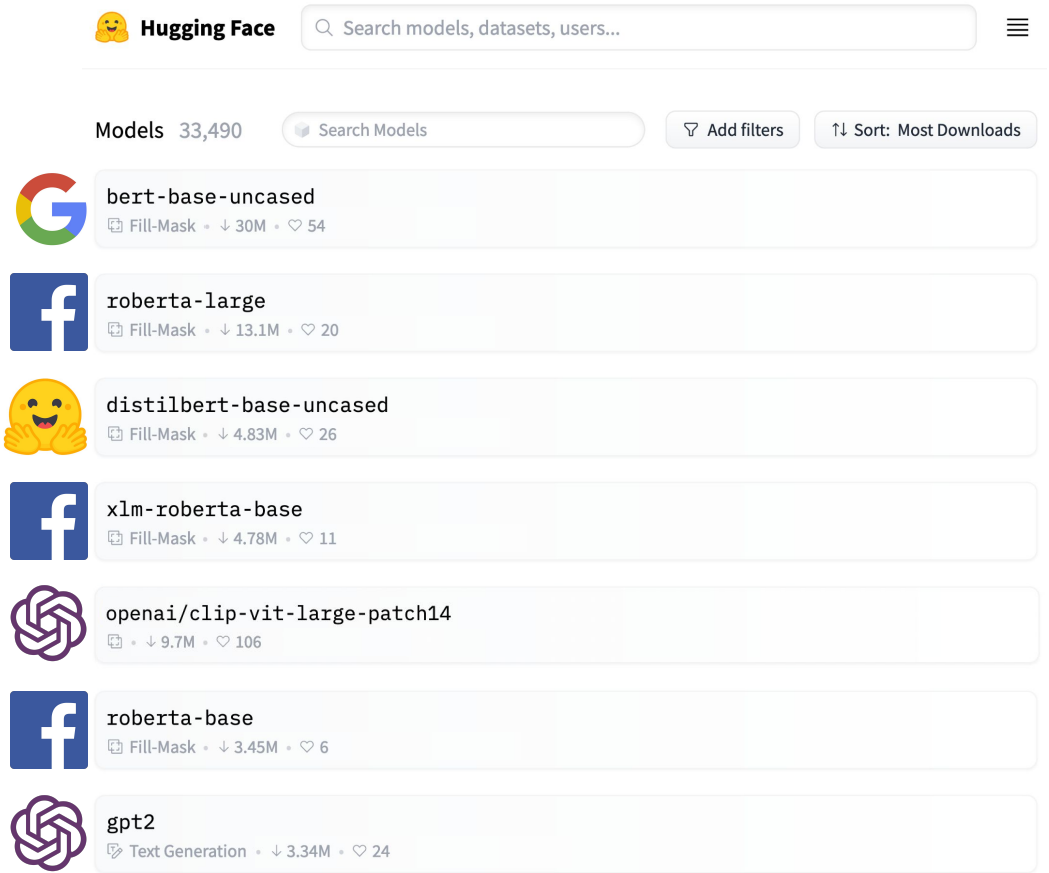




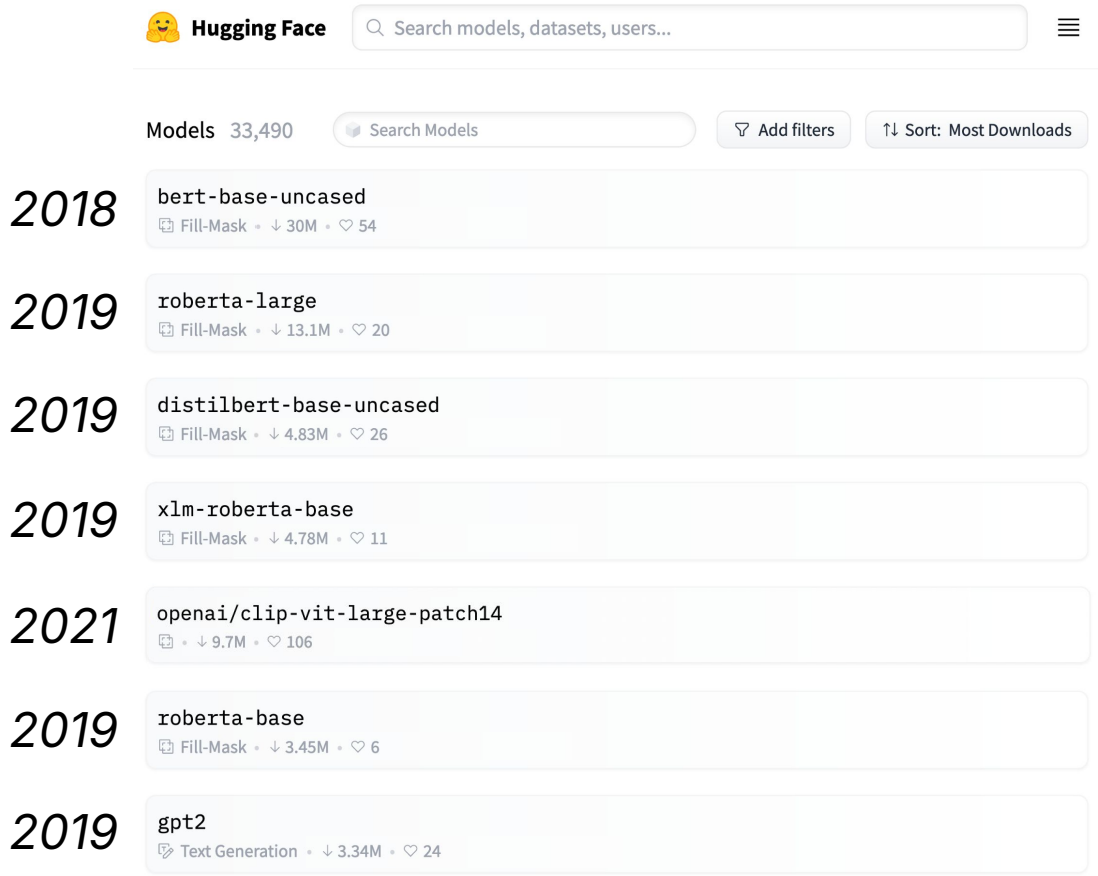
OpenAI technology, just an HTTPS call away

Apply our API to any language task — semantic search, summarization, sentiment analysis, content generation, translation, and more — with only a few examples or by specifying your task in English.

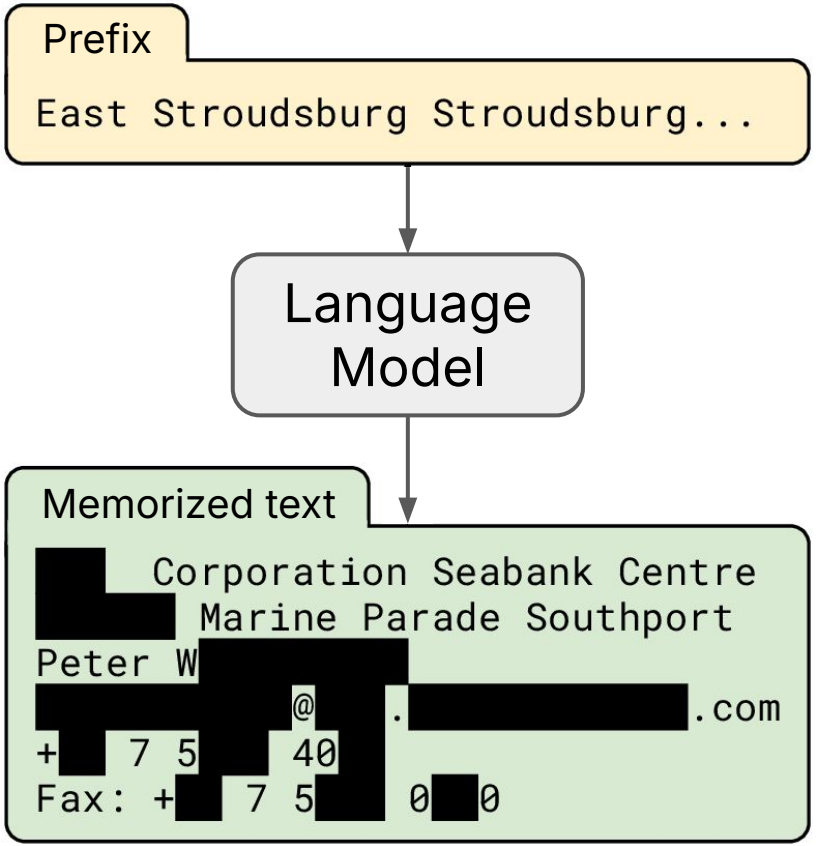
Popular public models often come from resource-rich groups



... and the models themselves are rarely updated

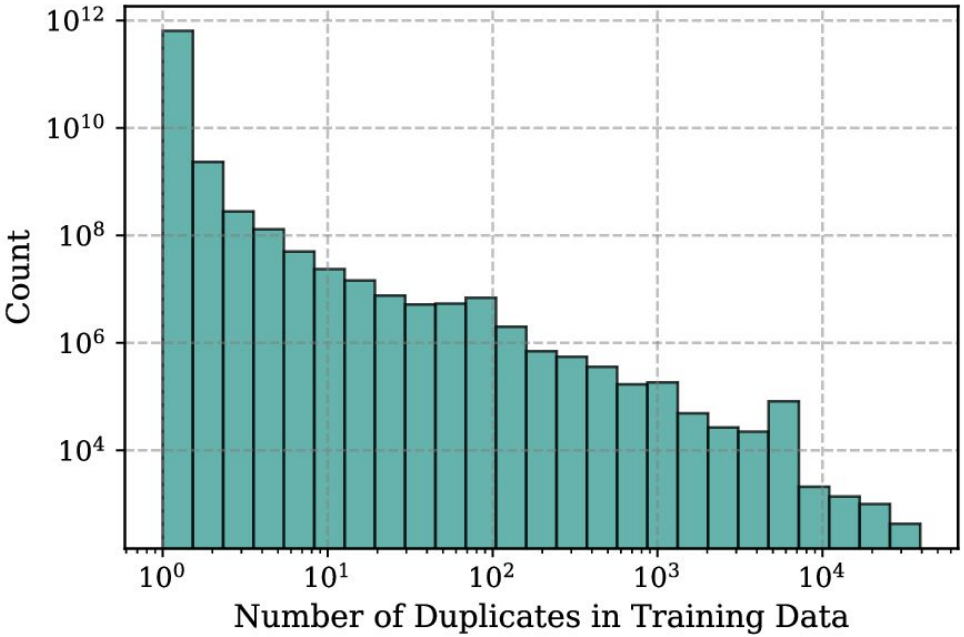


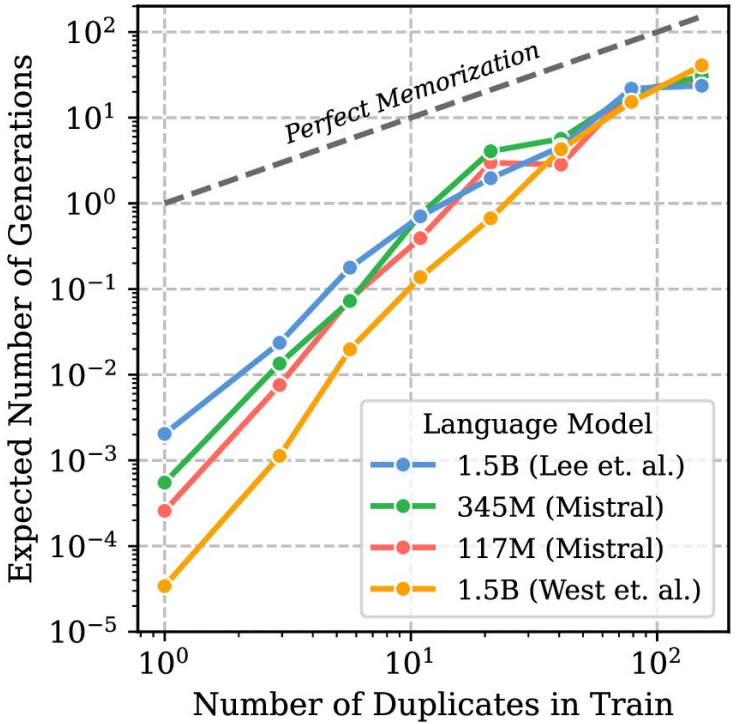
Models can exhibit issues, like memorized training data



From "Extracting Training Data from Large Language Models" by Carlini et al.

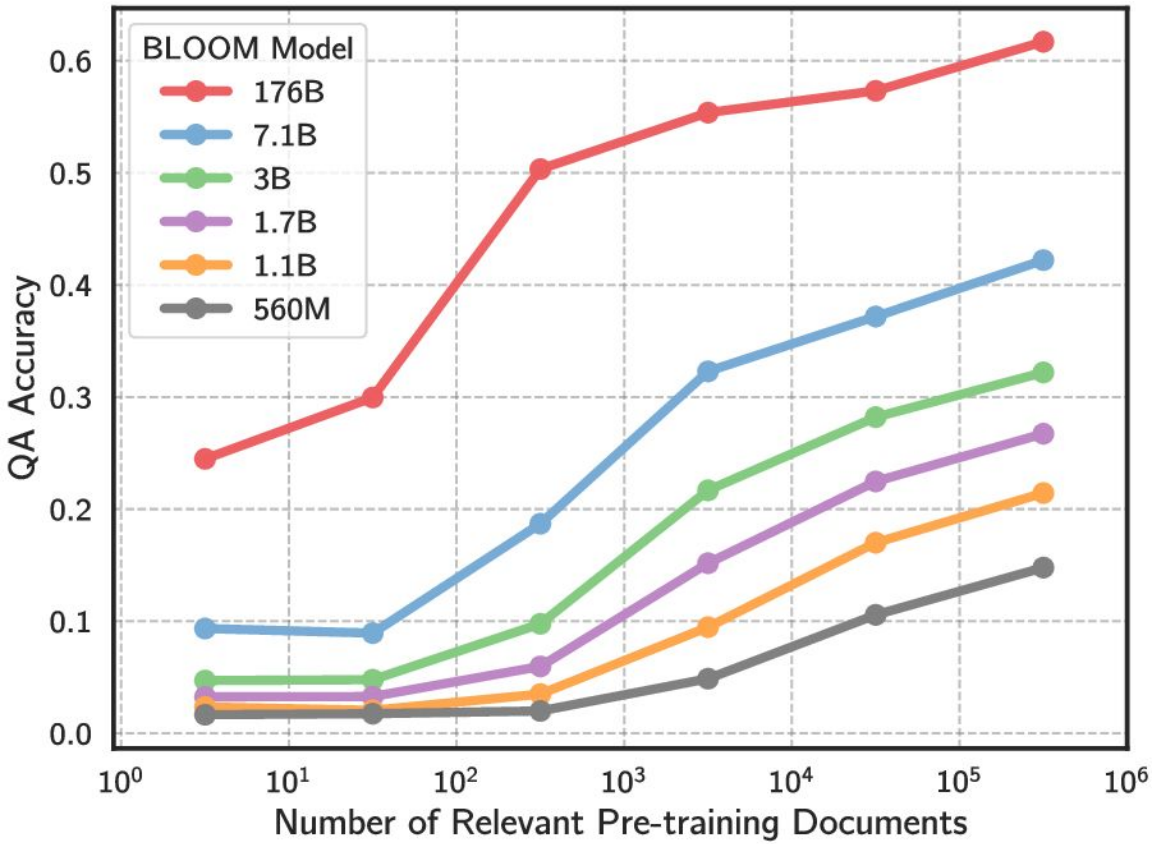
Issues with a model can be caused by issues with a dataset





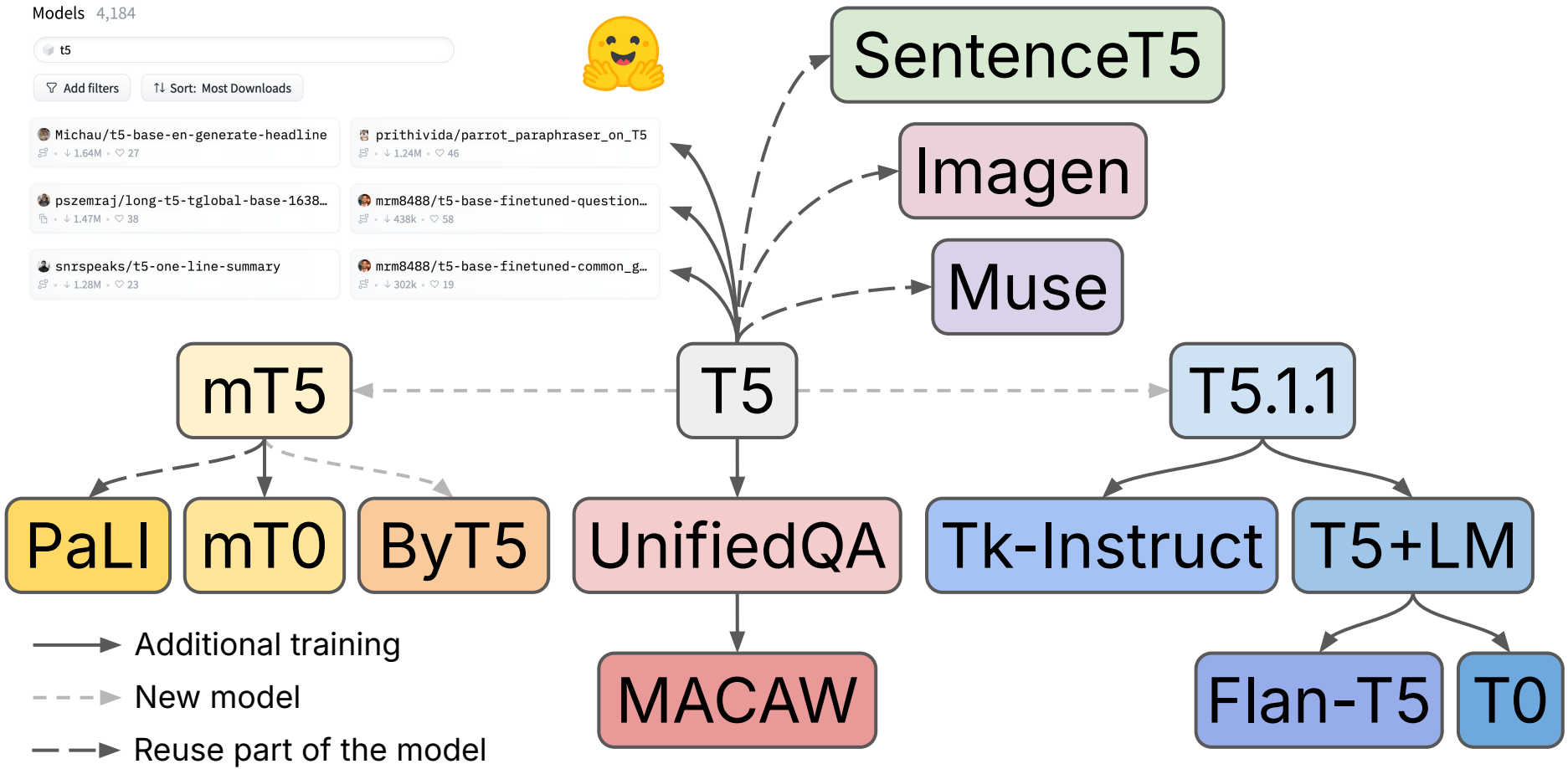
From "Deduplicating Training Data Mitigates Privacy Risks in Language Models" by Kandpal et al.

Pre-training datasets can also fail to address downstream needs



From "Large Language Models Struggle to Learn Long-Tail Knowledge" by Kandpal et al.

Pre-trained models are often used as the basis for derivative models



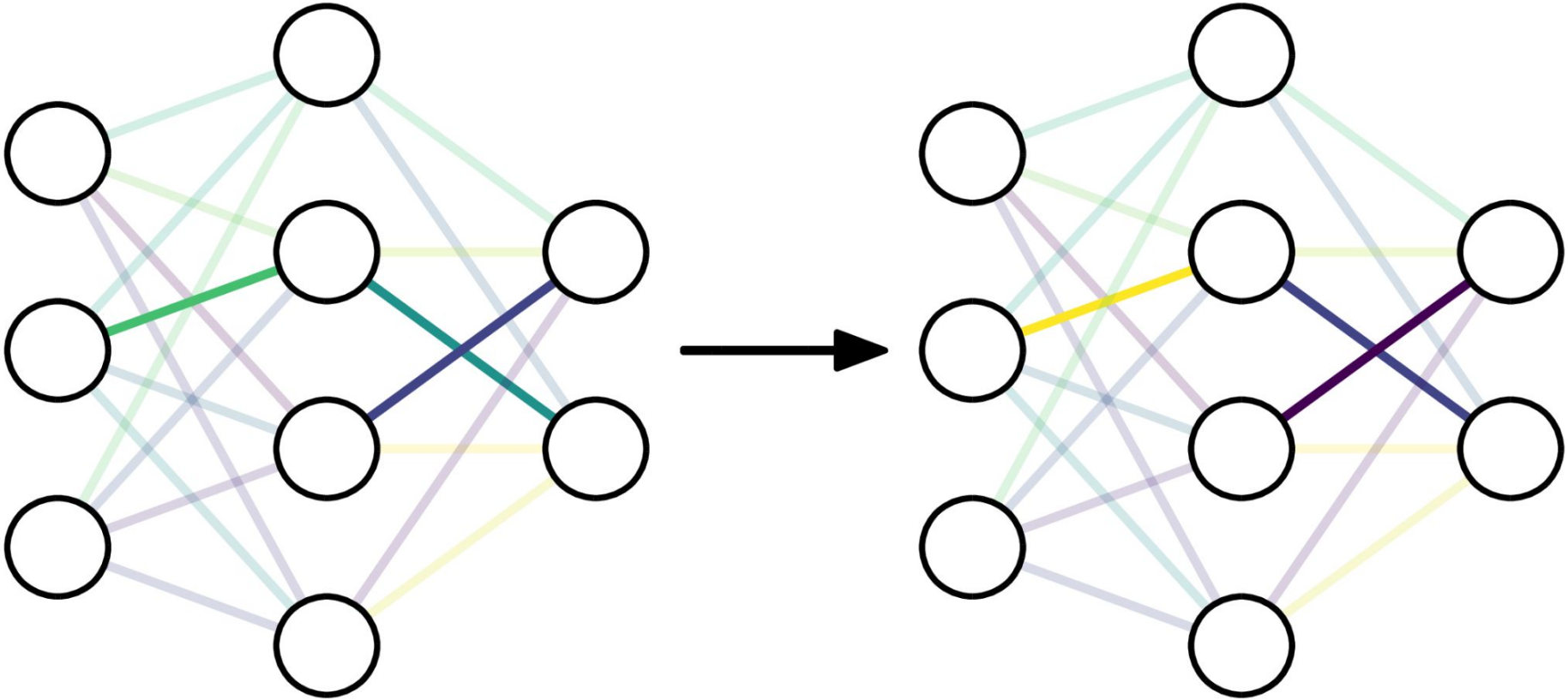

GCC
7ZIP
MySQL
RAILS
Open
SSL
TOP
SECRET
R

How can we enable collaborative and continual development of machine learning models?

How can we enable collaborative and continual development of machine learning models?

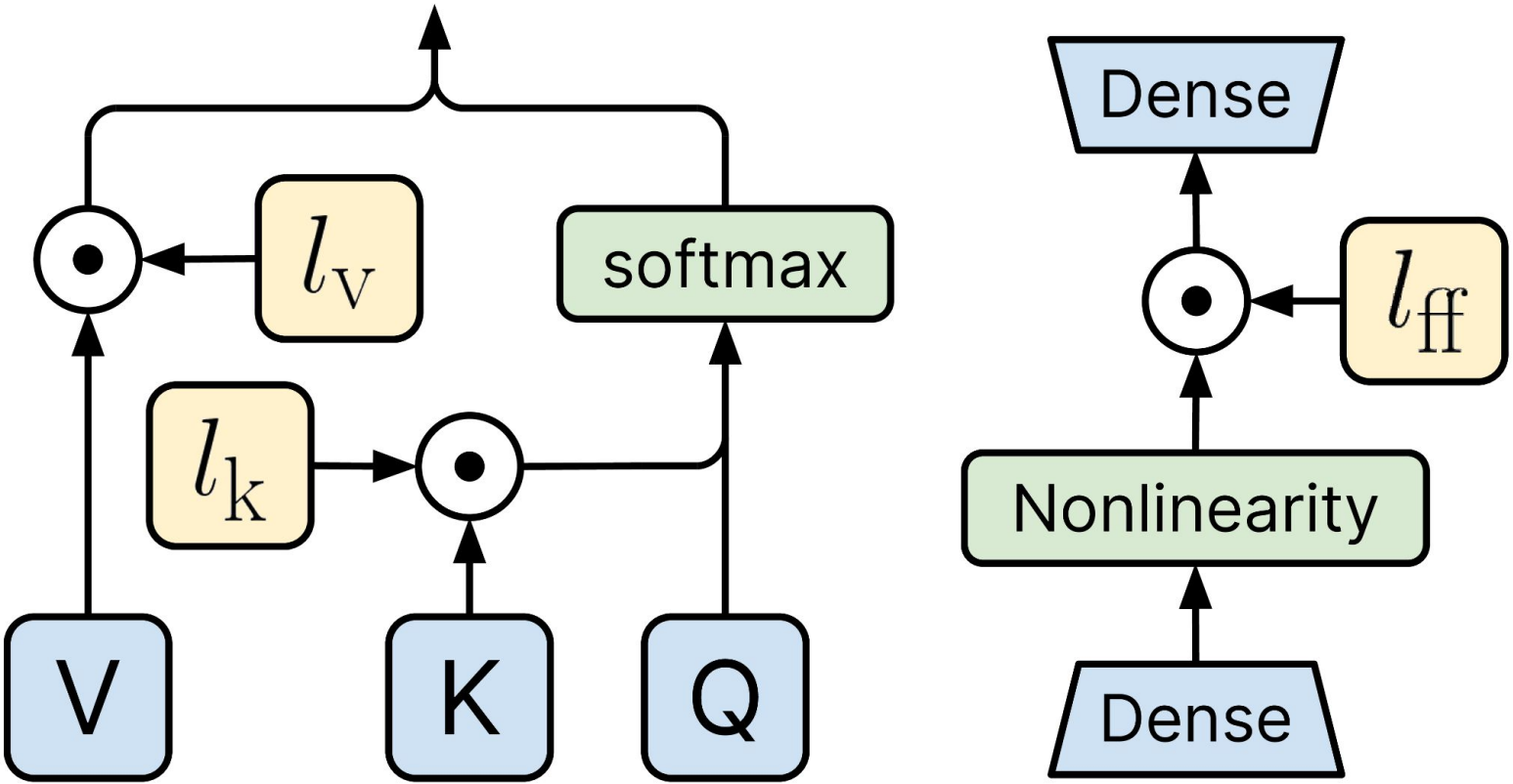
Contributors need to be able to cheaply communicate **patches** to a model.

Updating a subset of parameters reduces communication costs



From “Training Neural Networks with Fixed Sparse Masks" by Sung et al.

Updating models by rescaling activations

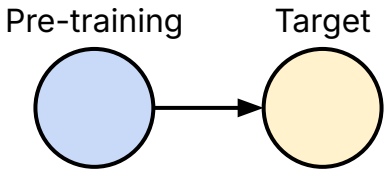


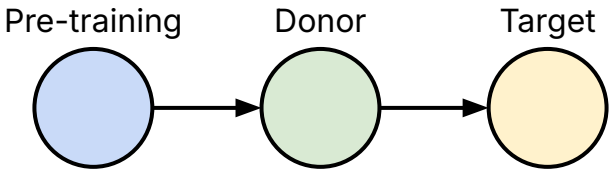
From "Few-Shot Parameter-Efficient Fine-Tuning is Better and Cheaper than In-Context Learning" by Liu et al.

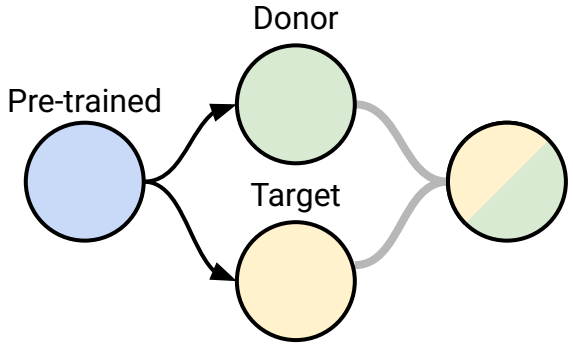
How can we enable collaborative and continual development of machine learning models?

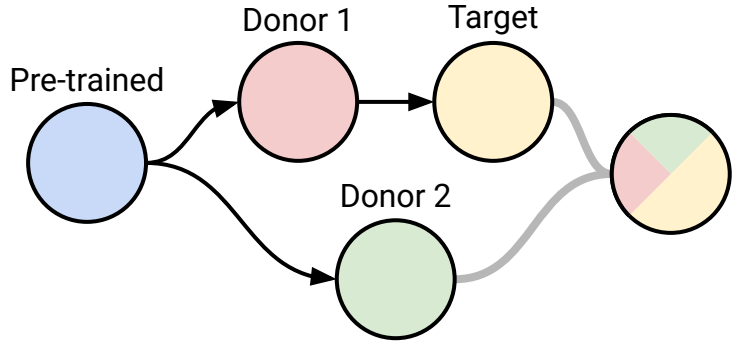
Maintainers need to be able to **merge** updates from different contributors.

Model merging enables new paths for transferring capabilities



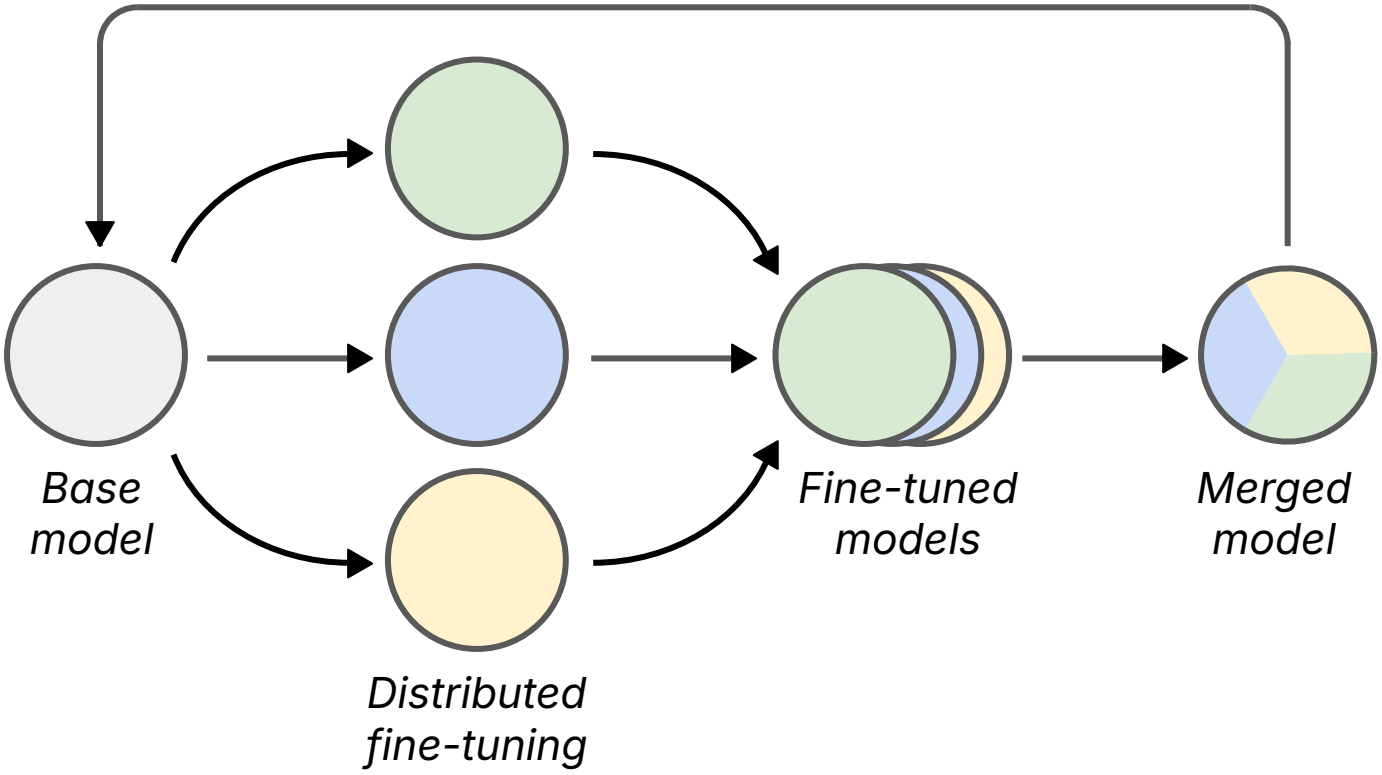






From "Merging Models with Fisher-Weighted Averaging" by Matena et al.

Merging fine-tuned models for better pre-trained models

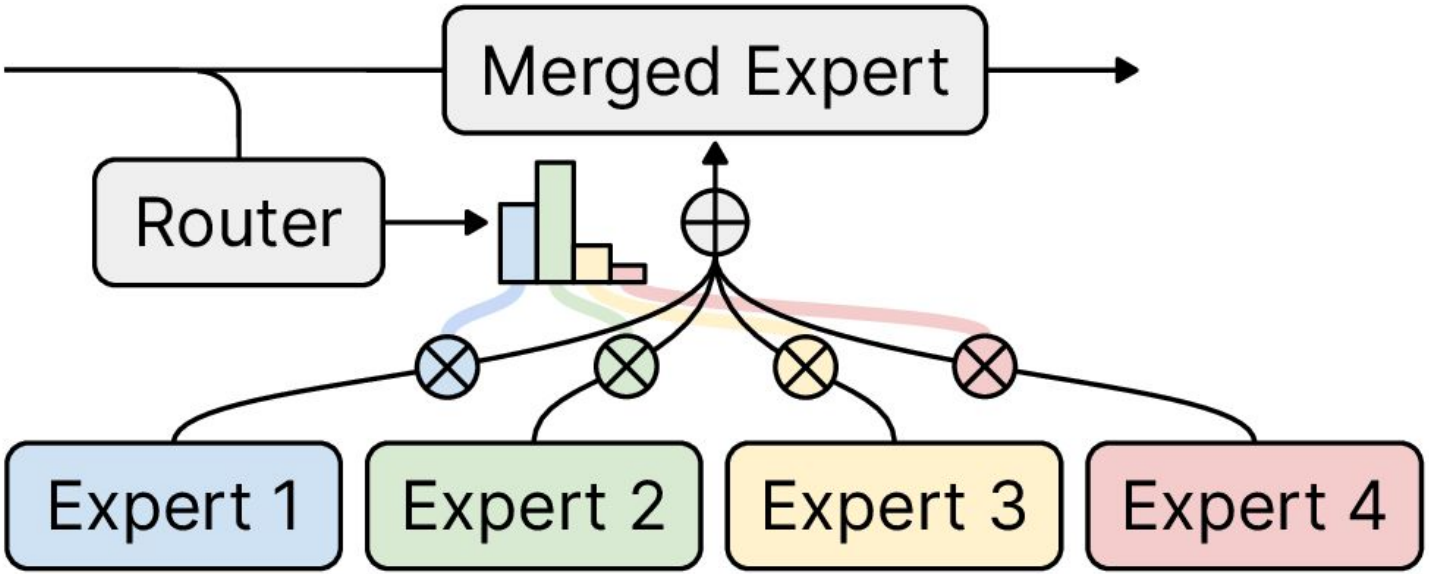


From "ColD Fusion: Collaborative Descent for Distributed Multitask Finetuning" by Don-Yehiya et al.

How can we enable collaborative and continual development of machine learning models?

We need to be able to combine **modular** components to enable new capabilities.

Modularity by merging experts with SMEAR

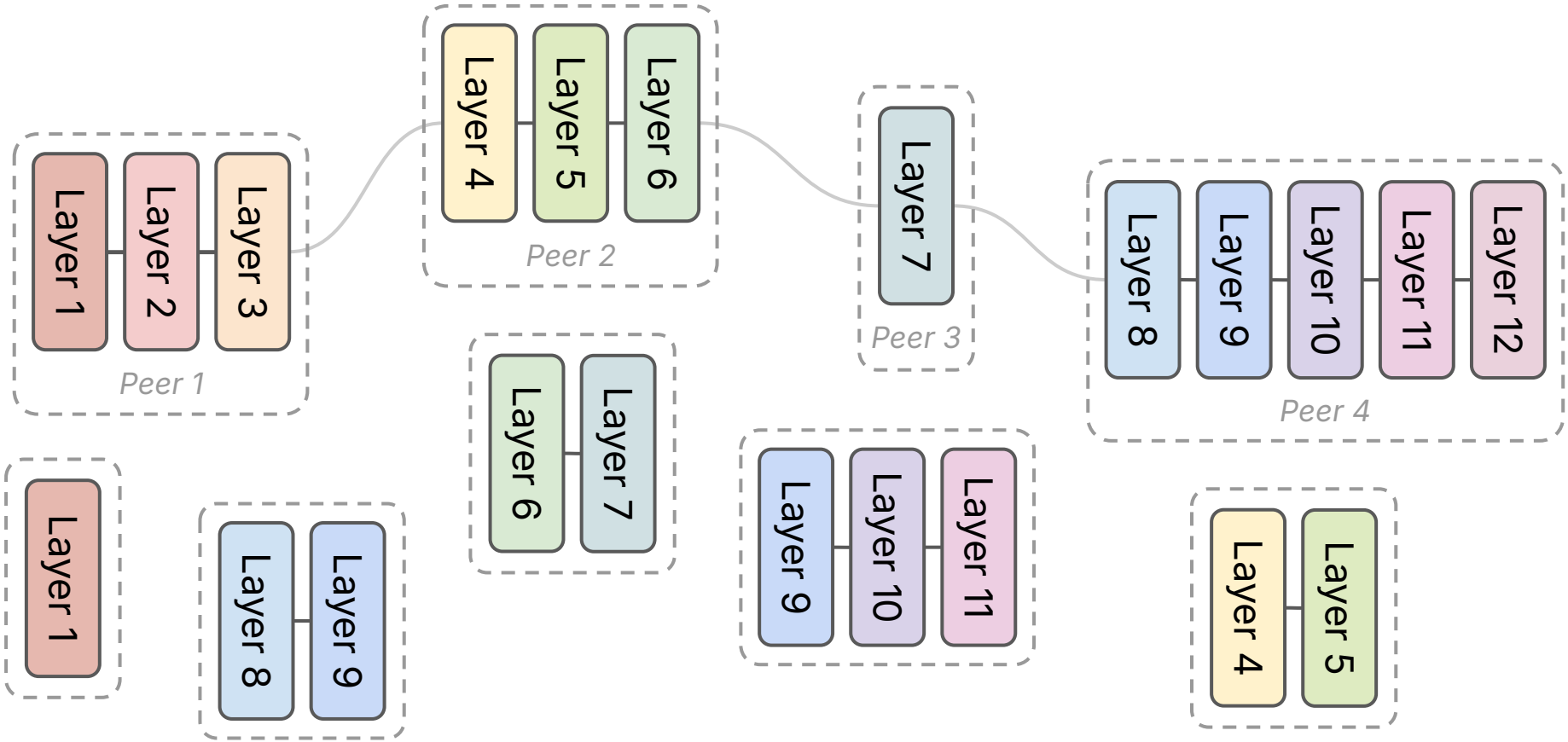


From “Soft Merging of Experts with Adaptive Routing" by Muqeeth et al.

How can we enable collaborative and continual development of machine learning models?

Users who lack resources need to be able to **train and run** large models.

PETALS enables distributed inference of large models over the internet



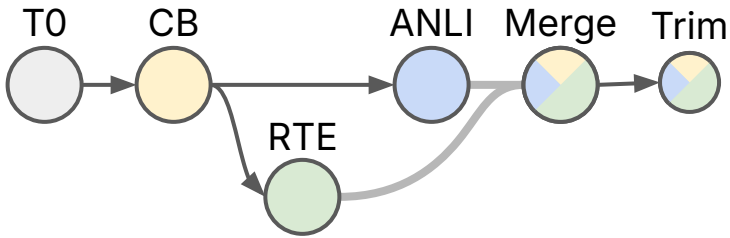
From "Petals: Collaborative Inference and Fine-tuning of Large Models" by Borzunov et al.

How can we enable collaborative and continual development of machine learning models?

We need a system for **version control** of model parameters.

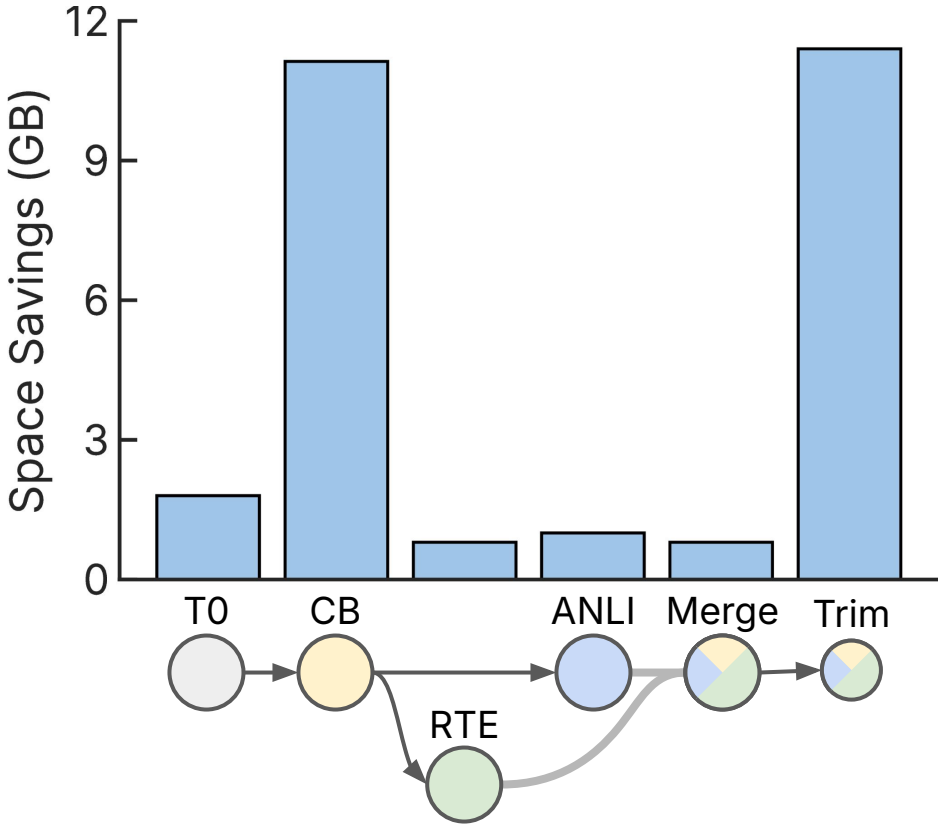
git-theta tracks, merges, and updates models using the git workflow

```
$ git-theta track model.pt
$ git commit -am "Add initial model"
$ python finetune.py --dataset="cb" --method="lowrank"
$ git commit -am "Fine-tune on CB dataset with LoRA"
$ git checkout -b rte
$ python finetune.py --dataset="rte" --method="dense"
$ git commit -am "Fine-tune on RTE dataset"
$ git checkout main
$ python finetune.py --dataset="anli" --method="dense"
$ git commit -am "Fine-tune on ANLI dataset"
$ git merge rte
Fixing Merge Conflicts in model.pt
Actions:
  avg)  average: Average parameter values.
  tt)  take_them: Use their change to the parameter.
  tu)  take_us: Use our change to the parameter.
  q)  quit
θ avg
$ git commit -am "Merge RTE and ANLI models"
$ python trim_unused_embeddings.py
$ git commit -am "Remove embeddings for unused tokens"
```



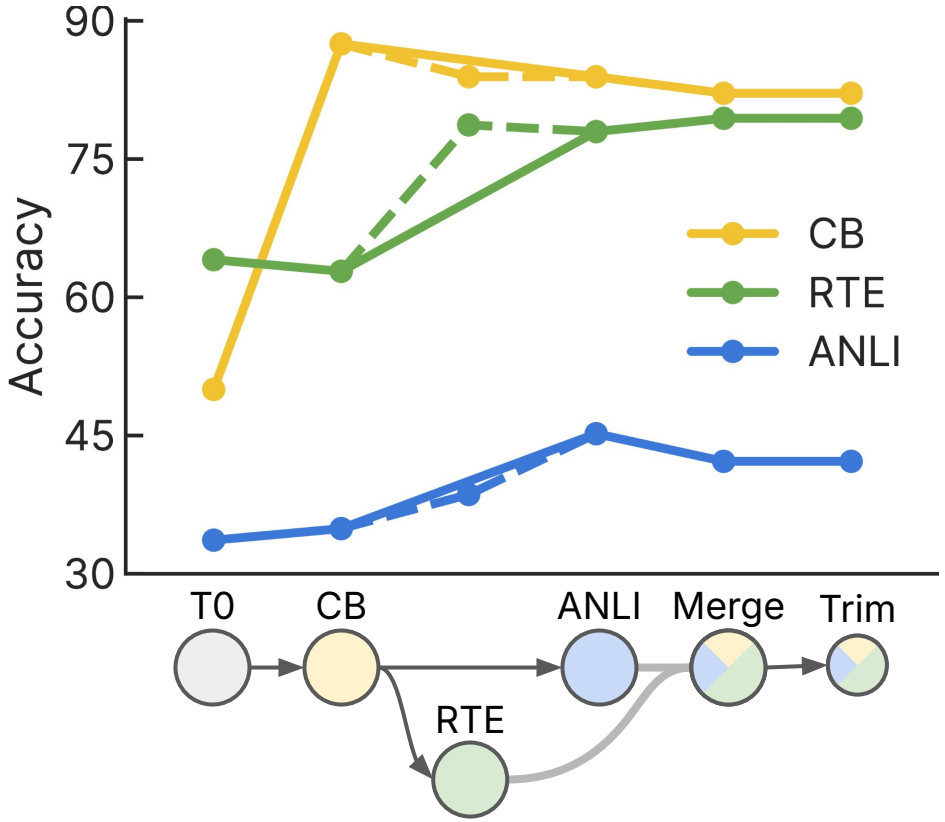
From "Git-Theta: A Git Extension for Collaborative Development of Machine Learning Models" by Kandpal et al.

Communication-efficient updates result in significant space savings



From "Git-Theta: A Git Extension for Collaborative Development of Machine Learning Models" by Kandpal et al.

git-theta allows for continuous and collaborative model development



From "Git-Theta: A Git Extension for Collaborative Development of Machine Learning Models" by Kandpal et al.

Building Machine Learning Models Like Open Source Software, *Communications of the ACM*
Colin Raffel

Extracting Training Data from Large Language Models, *USENIX Security 2021*
Nicholas Carlini, Florian Tramer, Eric Wallace, Matthew Jagielski, Ariel Herbert-Voss, Katherine Lee, Adam Roberts, Tom Brown, Dawn Song, Ulfar Erlingsson, Alina Oprea, & **Colin Raffel**

Deduplicating Training Data Mitigates Privacy Risks in Language Models, *ICML* 2022
Nikhil Kandpal, Eric Wallace, & **Colin Raffel**

Large Language Models Struggle to Learn Long-Tail Knowledge, *ICML 2023*
Nikhil Kandpal, Haikang Deng, Adam Roberts, Eric Wallace, & **Colin Raffel**

Training Neural Networks with Fixed Sparse Masks, *NeurIPS 2021*
Yi-Lin Sung*, Varun Nair*, & **Colin Raffel**

Few-Shot Parameter-Efficient Fine-Tuning is Better and Cheaper than In-Context Learning, *NeurIPS 2022*
Haokun Liu*, Derek Tam*, Mohammed Muqeeth*, Jay Mohta, Tenghao Huang, Mohit Bansal, & **Colin Raffel**

Merging Models with Fisher-Weighted Averaging, *NeurIPS 2022*
Michael Matena & **Colin Raffel**

ColD Fusion: Collaborative Descent for Distributed Multitask Finetuning, *in submission*
Shachar Don-Yehiya, Elad Venezian, **Colin Raffel**, Noam Slonim, Yoav Katz, & Leshem Choshen

Soft Merging of Experts with Adaptive Routing, *in submission*
Mohammed Muqeeth, Haokun Liu, & **Colin Raffel**

Git-Theta: A Git Extension for Collaborative Development of Machine Learning Models, *ICML 2023*
Nikhil Kandpal*, Brian Lester*, Mohammed Muqeeth, Anisha Mascarenhas, Monty Evans, Vishal Baskaran, Tenghao Huang, Haokun Liu, & **Colin Raffel**

Petals: Collaborative Inference and Fine-tuning of Large Models, *ACL 2023*
Alexander Borzunov, Dmitry Baranchuk, Tim Dettmers, Max Ryabinin, Younes Belkada, Artem Chumachenko, Pavel Samygin, & **Colin Raffel**