SILO Open LM:
Training LMs on Siloed Datasets

Sewon Min

sewonmin.com

Berkeley EECS

Ai2

Presenting work with

Weijia Shi, Akshita Bhagia, Kevin Farhat

+ Other collaborators at Ai2!

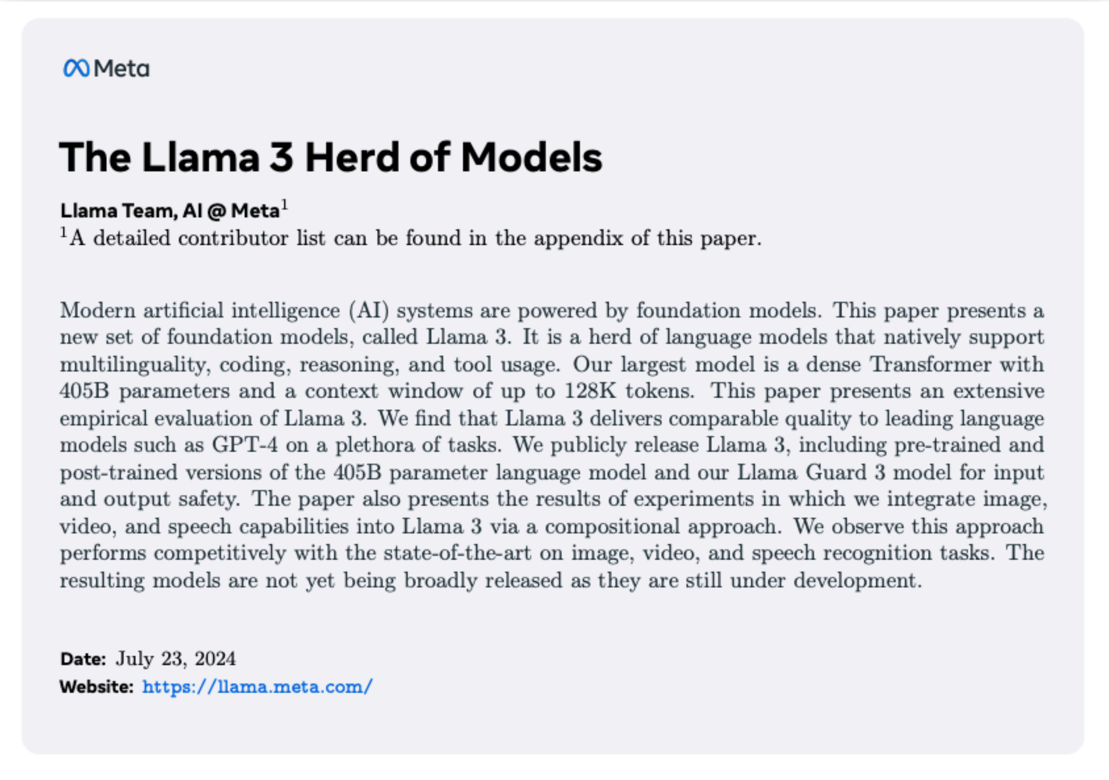
Common ground

Common ground

- Data is very important

Common ground

- Data is very important
- Data is also secret



Meta

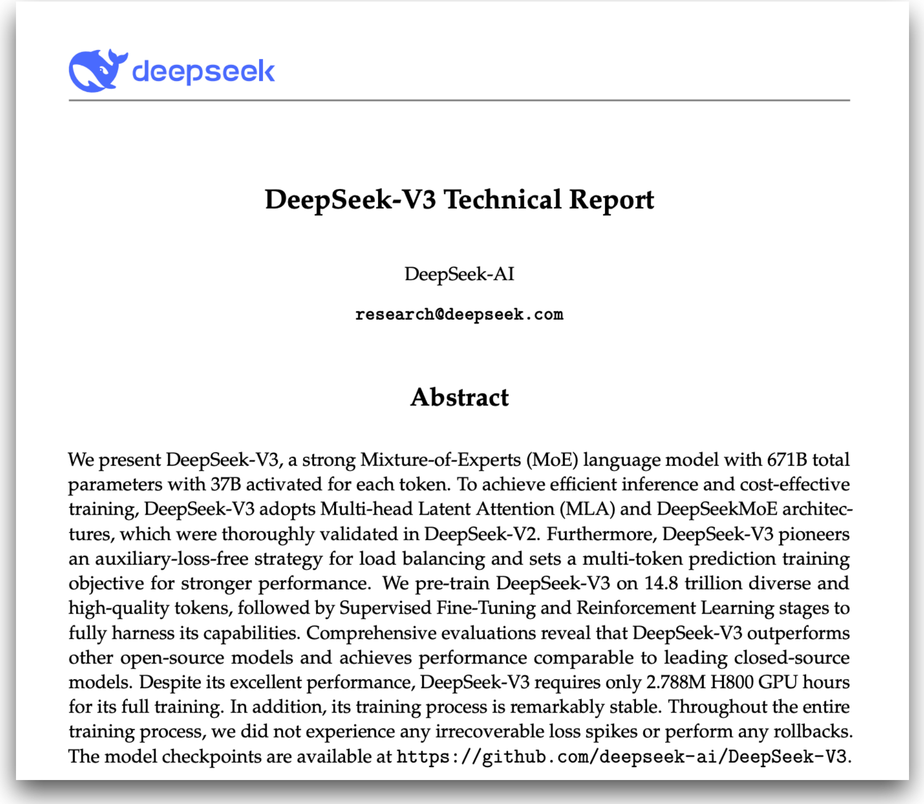
The Llama 3 Herd of Models

Llama Team, AI @ Meta[1]

[1]A detailed contributor list can be found in the appendix of this paper.

Modern artificial intelligence (AI) systems are powered by foundation models. This paper presents a new set of foundation models, called Llama 3. It is a herd of language models that natively support multilinguality, coding, reasoning, and tool usage. Our largest model is a dense Transformer with 405B parameters and a context window of up to 128K tokens. This paper presents an extensive empirical evaluation of Llama 3. We find that Llama 3 delivers comparable quality to leading language models such as GPT-4 on a plethora of tasks. We publicly release Llama 3, including pre-trained and post-trained versions of the 405B parameter language model and our Llama Guard 3 model for input and output safety. The paper also presents the results of experiments in which we integrate image, video, and speech capabilities into Llama 3 via a compositional approach. We observe this approach performs competitively with the state-of-the-art on image, video, and speech recognition tasks. The resulting models are not yet being broadly released as they are still under development.

Date: July 23, 2024
Website: https://llama.meta.com/



deepseek

DeepSeek-V3 Technical Report

DeepSeek-AI

research@deepseek.com

Abstract

We present DeepSeek-V3, a strong Mixture-of-Experts (MoE) language model with 671B total parameters with 37B activated for each token. To achieve efficient inference and cost-effective training, DeepSeek-V3 adopts Multi-head Latent Attention (MLA) and DeepSeekMoE architectures, which were thoroughly validated in DeepSeek-V2. Furthermore, DeepSeek-V3 pioneers an auxiliary-loss-free strategy for load balancing and sets a multi-token prediction training objective for stronger performance. We pre-train DeepSeek-V3 on 14.8 trillion diverse and high-quality tokens, followed by Supervised Fine-Tuning and Reinforcement Learning stages to fully harness its capabilities. Comprehensive evaluations reveal that DeepSeek-V3 outperforms other open-source models and achieves performance comparable to leading closed-source models. Despite its excellent performance, DeepSeek-V3 requires only 2.788M H800 GPU hours for its full training. In addition, its training process is remarkably stable. Throughout the entire training process, we did not experience any irrecoverable loss spikes or perform any rollbacks. The model checkpoints are available at `https://github.com/deepseek-ai/DeepSeek-V3`.

Common ground

- Data is very important
- Data is also secret
- (Valuable) data is proprietary

News, textbooks, educational contents, etc

Common ground

- Data is very important
- Data is also secret
- (Valuable) data is proprietary

  Or, becoming proprietary, even when it used not to

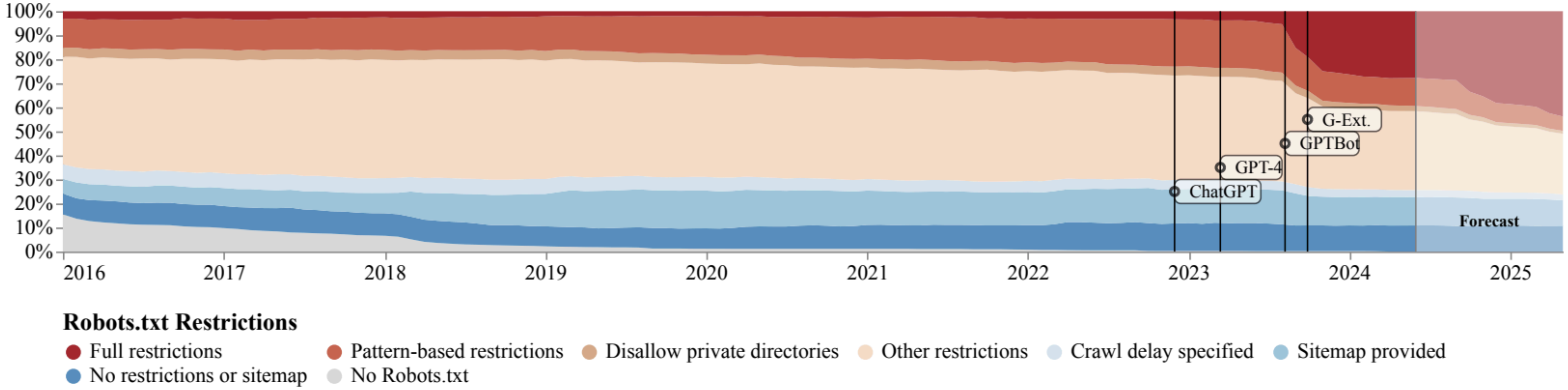
News, textbooks, educational contents, etc

Reddit locks down its public data in new content policy, says use now requires a contract

Stack Overflow Will Charge AI Giants for Training Data

The programmer Q&A site joins Reddit in demanding compensation when its data is used to train

Common ground



Longpre et al. 2024. "Consent in Crisis: The Rapid Decline of the AI Data Commons"

How can research community make progress?

How can research community make progress?

1. Develop open-source training data that is publicly available

Collection by mlfoundations

DCLM

DCLM Models + Datasets

togethercomputer/
RedPajama-Data

How can research community make progress?

1. Develop open-source training data that is publicly available

OLMo

Collection by mlfoundations

DCLM

DCLM Models + Datasets

togethercomputer/
RedPajama-Data

2. Design new model architectures & training methods accommodating this data landscape

How can research community make progress?

1. Develop open-source training data that is publicly available

Collection by mlfoundations

DCLM

DCLM Models + Datasets

togethercomputer/
RedPajama-Data

2. Design new model architectures & training methods accommodating this data landscape

Today's talk!

So far, we have assumed the access to the data is binary -- either available, or not available


Available
Unavailable

- Data can’t be shared
- Model trained on the data can be shared

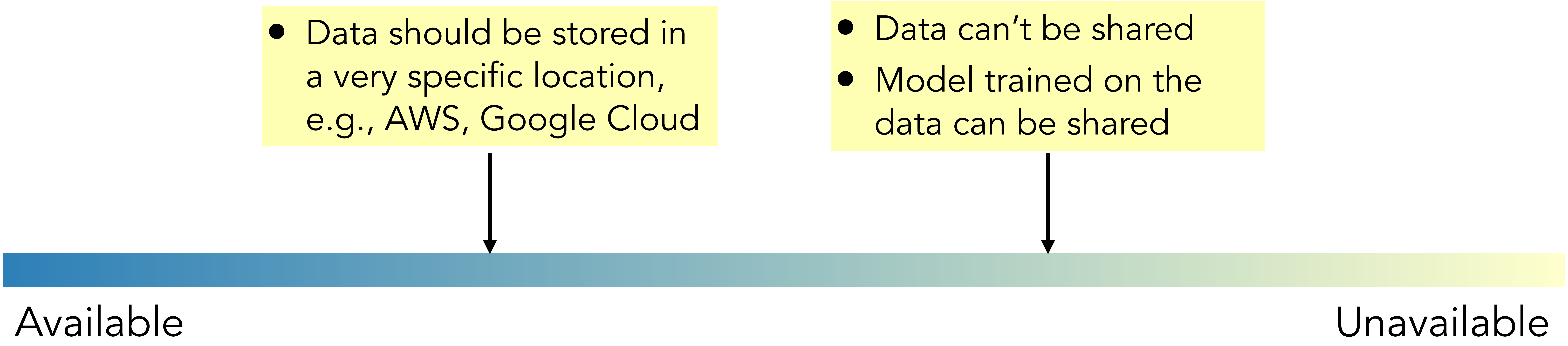
Available Unavailable

- Data should be stored in a very specific location, e.g., AWS, Google Cloud

- Data can't be shared
- Model trained on the data can be shared

Available

Unavailable



Other Restrictions: Data may become available at different times, may have expiration dates or owner might require opt-out.

- Data should be stored in a very specific location, e.g., AWS, Google Cloud

- Data can't be shared
- Model trained on the data can be shared

Available

Unavailable

*Data should be located in the same place, we can take a union and randomly shuffle the data, all data is available at the same time, all data is available forever (no expiration, no opt-out request), etc.…

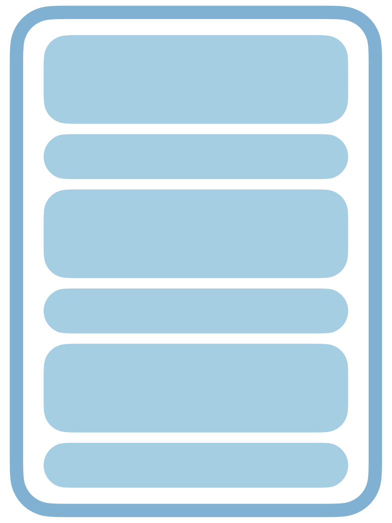
Other Restrictions: Data may become available at different times, may have expiration dates or owner might require opt-out.

When we can't train on all datasets jointly,
how can we train a general-purpose LM that
still benefits from ***siloed*** datasets?

Setup

COMMON CRAWL

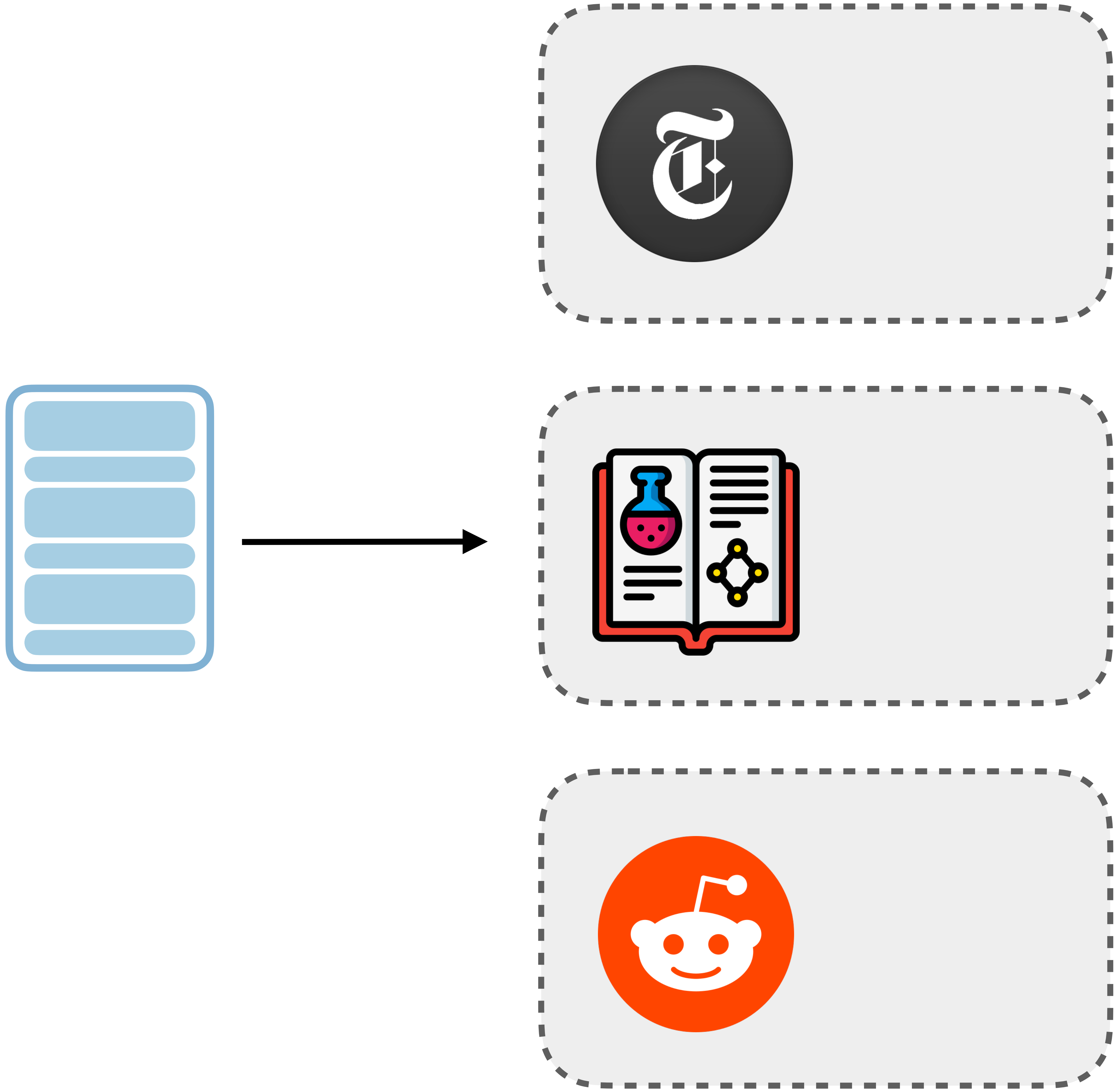
Public, shared data



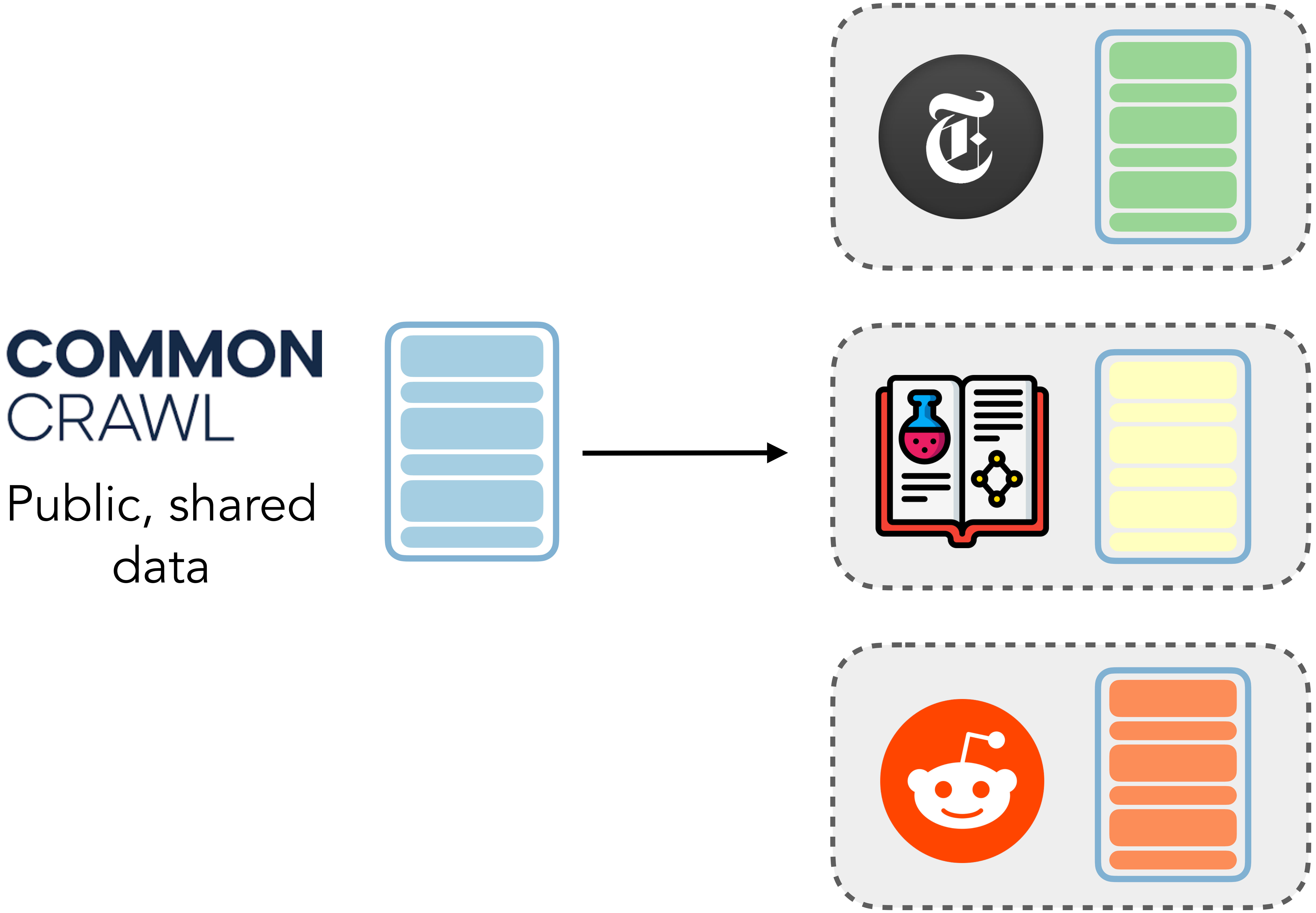
Setup

COMMON CRAWL

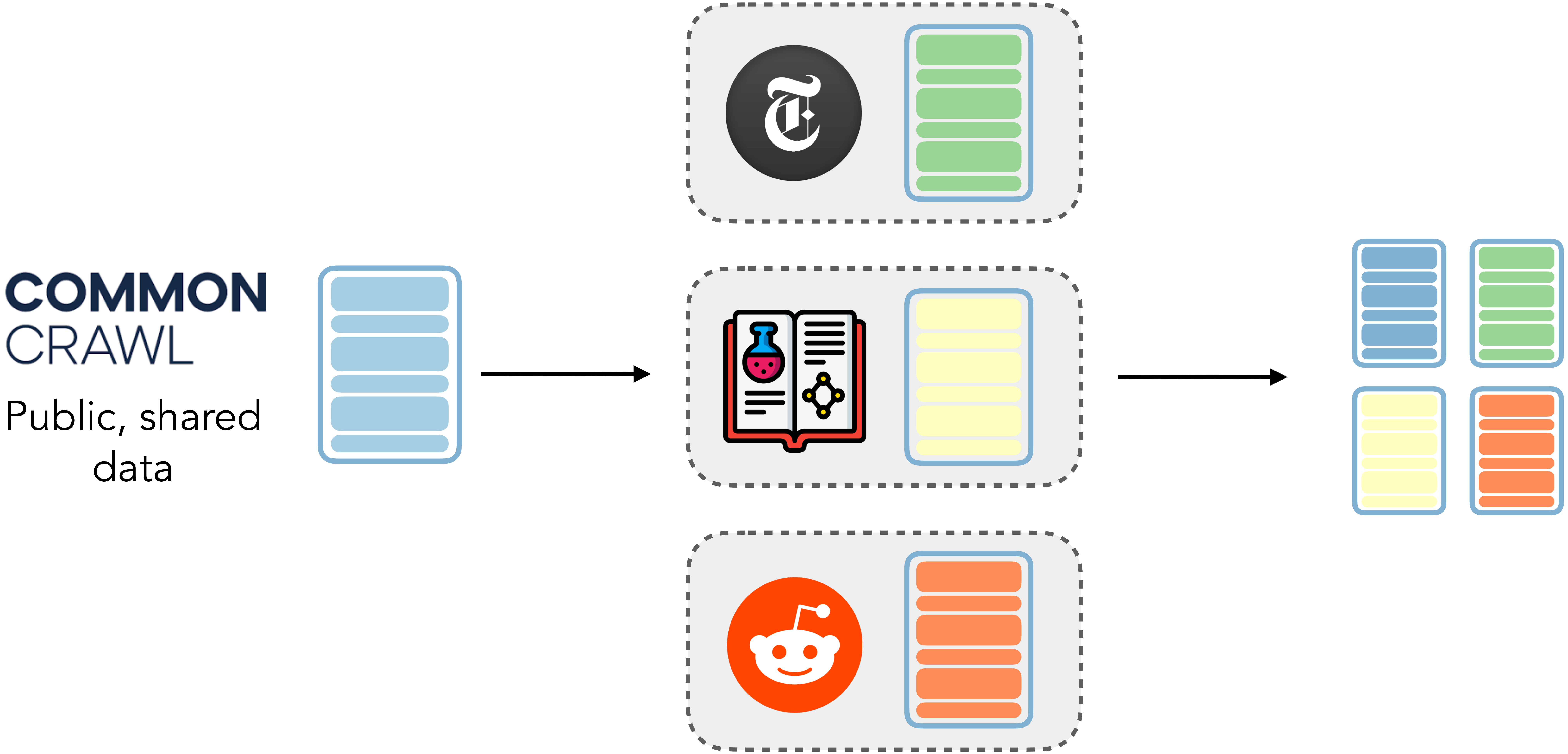
Public, shared data



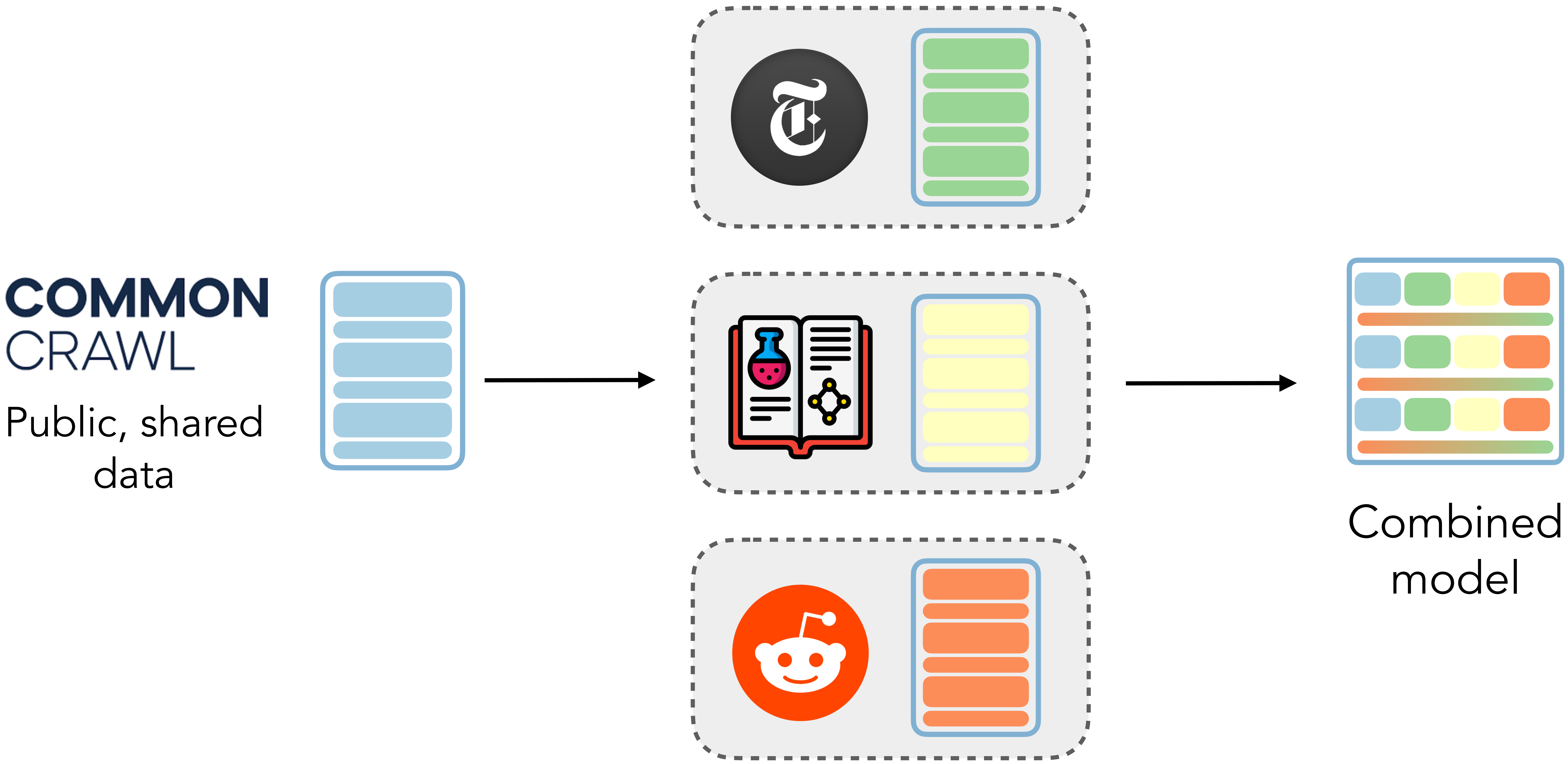
Setup



Setup



Setup



Requirements

- Train on *isolated*, *distributed* datasets **(siloed datasets)**
- Train in isolation
- Resulting model should be a public, general-purpose model
- Support data opt-out, free/cheap addition/removal

Requirements

- Train on *isolated, distributed* datasets **(siloed datasets)**
- Train in isolation
- Resulting model should be a public, general-purpose model
- Support data opt-out, free/cheap addition/removal

Federated learning satisfies the first requirement but not others

Experimental setup

COMMON
CRAWL

Public data

Experimental setup

COMMON CRAWL

Public data

Math

Code

Creative writings

Papers

News

Reddit

Textbooks

Experimental setup

COMMON
CRAWL

Public data

Math

Code

Creative writings

Papers

News

Reddit

Textbooks

← Actual proprietary data (Can't publicly acquire as of now)

Experimental setup

COMMON CRAWL

Public data

Math

Code

Creative writings

Papers

← Simulated proprietary data (Realistic domains)

News

Reddit

Textbooks

How to design models?

Related existing methods

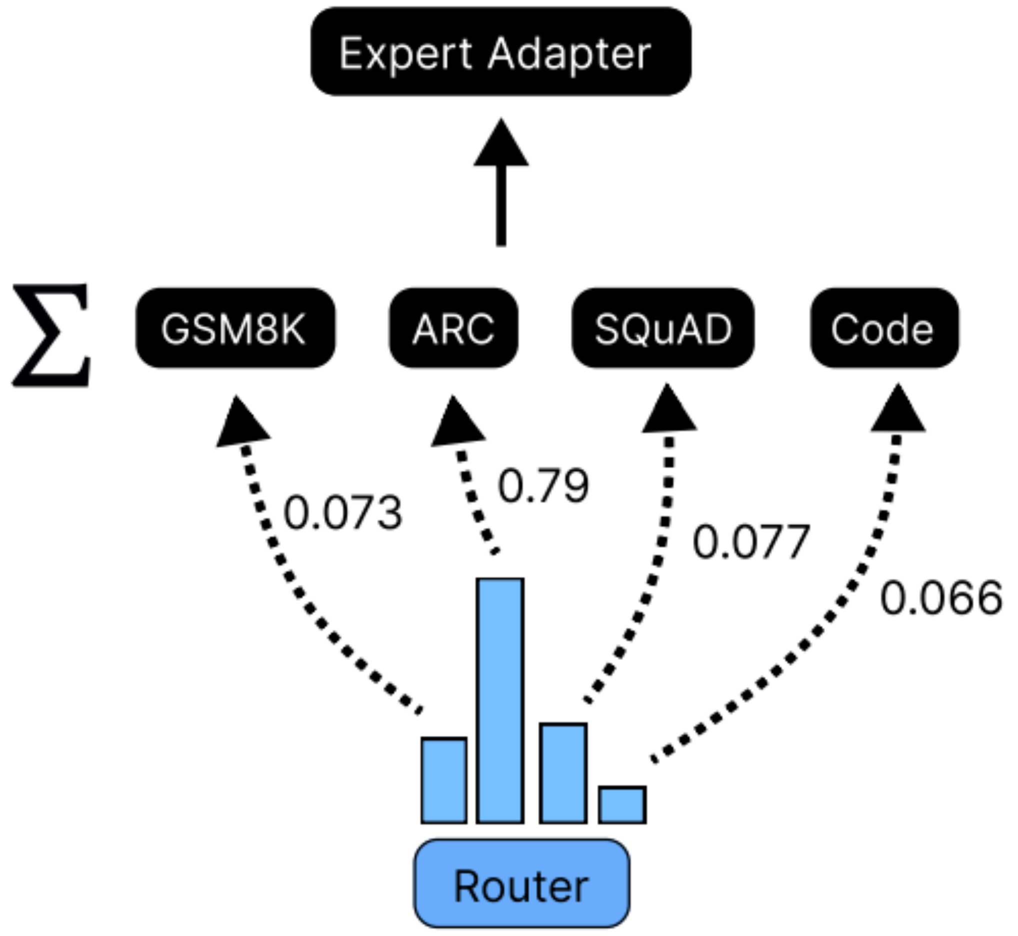
Task fine-tuning

Pre-training

Related existing methods

Task fine-tuning

- Setup was explored, methods are not applicable



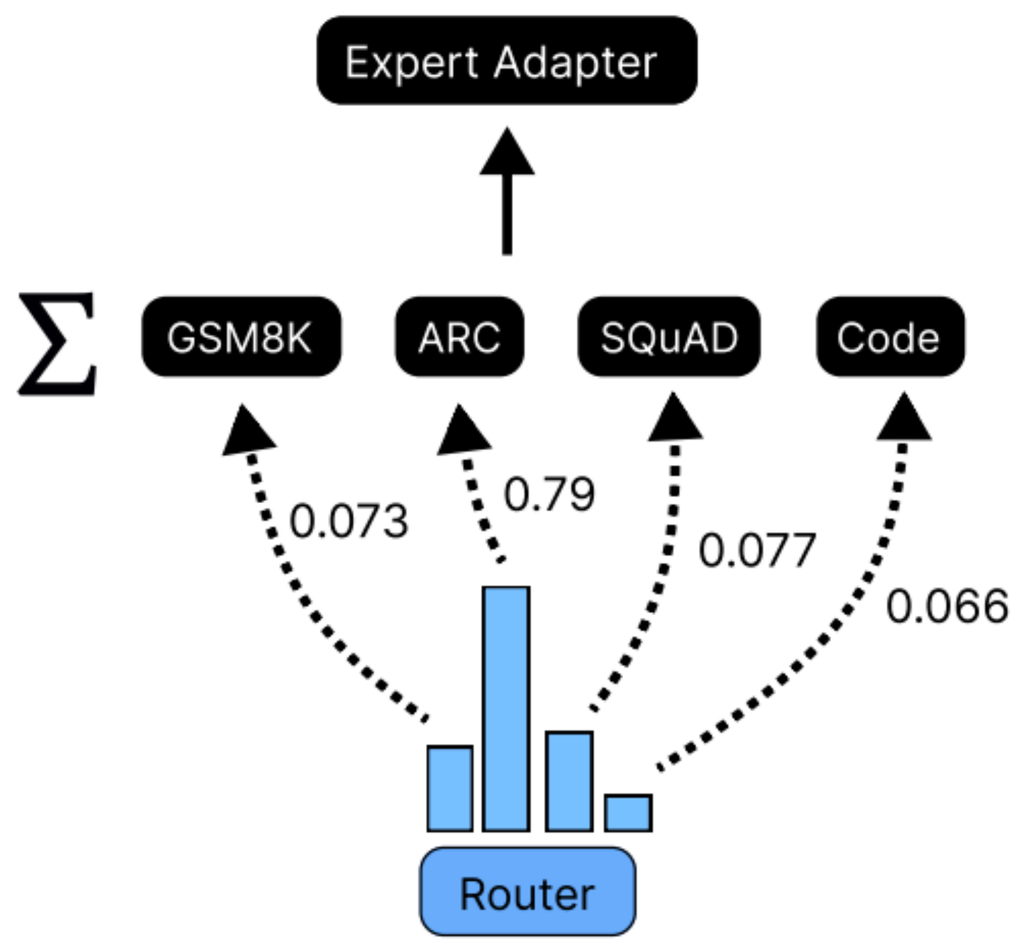
Pre-training

Yadav & Raffel et al. 2024. "A Survey on Model MoErging: Recycling and Routing Among Specialized Experts for Collaborative Learning"
Buehler & Buehler, 2024. "X-LoRA: Mixture of Low-Rank Adapter Experts, a Flexible Framework for Large Language Models …"
Jang et al. 2023. "Exploring the benefits of training expert language models over instruction tuning"
Belofsky. 2023. "Token-level adaptation of lora adapters for downstream task generalization"

Related existing methods

Task fine-tuning

- Setup was explored, methods are not applicable



Pre-training

- Setup was underexplored
- Model merging for boosting performance or for specialization has been explored (next slides)

Yadav & Raffel et al. 2024. "A Survey on Model MoErging: Recycling and Routing Among Specialized Experts for Collaborative Learning"
Buehler & Buehler, 2024. "X-LoRA: Mixture of Low-Rank Adapter Experts, a Flexible Framework for Large Language Models …"
Jang et al. 2023. "Exploring the benefits of training expert language models over instruction tuning"
Belofsky. 2023. "Token-level adaptation of lora adapters for downstream task generalization"

Existing approach 1: Weight merging/ensembling

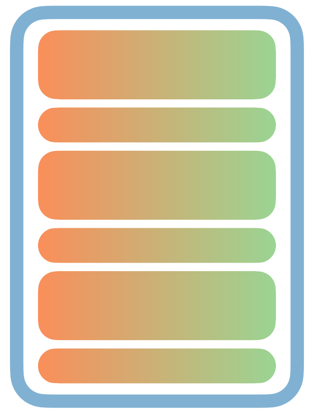
How to merge *n* different LMs, $w_0, w_1, \cdots, w_{n-1}$, to boost performance?

Li et al. 2022. “Branch-Train-Merge: Embarrassingly Parallel Training of Expert Language Models”
Gururangan et al. 2023. "Scaling Expert Language Models with Unsupervised Domain Discovery”

Existing approach 1: Weight merging/ensembling

How to merge *n* different LMs, $w_0, w_1, \cdots, w_{n-1}$, to boost performance?

Weight merging



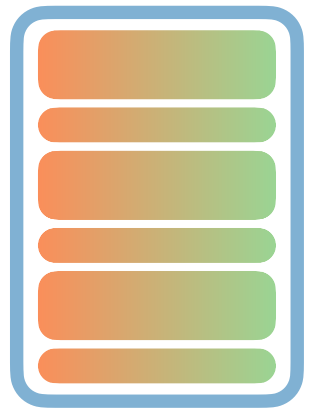
$$y = f(\frac{1}{n}\sum_{i=0}^{n-1} w_i, x)$$

Li et al. 2022. “Branch-Train-Merge: Embarrassingly Parallel Training of Expert Language Models”
Gururangan et al. 2023. "Scaling Expert Language Models with Unsupervised Domain Discovery”

Existing approach 1: Weight merging/ensembling

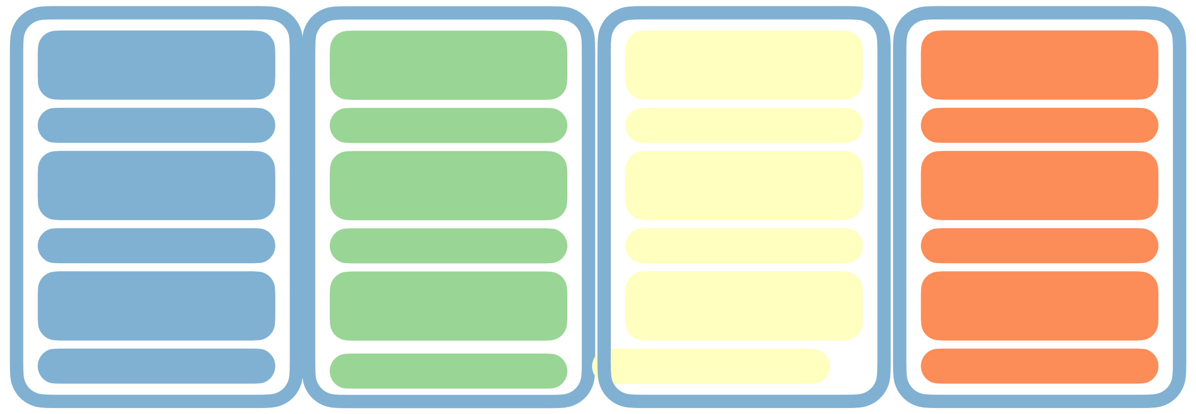
How to merge n different LMs, $w_0, w_1, \cdots, w_{n-1}$, to boost performance?

Weight merging



$$y = f(\frac{1}{n}\sum_{i=0}^{n-1} w_i, x)$$

Ensembling



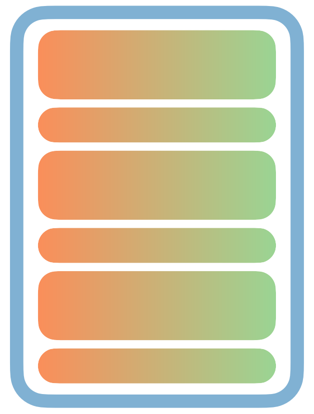
$$y = \frac{1}{n}\sum_{i=0}^{n-1} f(w_i, x)$$

Li et al. 2022. "Branch-Train-Merge: Embarrassingly Parallel Training of Expert Language Models"
Gururangan et al. 2023. "Scaling Expert Language Models with Unsupervised Domain Discovery"

Existing approach 1: Weight merging/ensembling

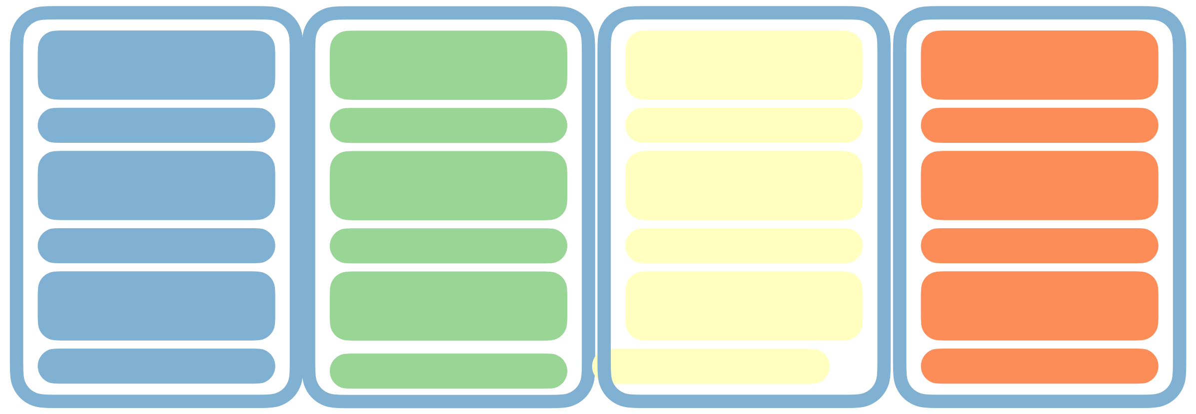
How to merge n different LMs, $w_0, w_1, \cdots, w_{n-1}$, to boost performance?

Weight merging



$$y = f(\frac{1}{n}\sum_{i=0}^{n-1} w_i, x)$$

Ensembling



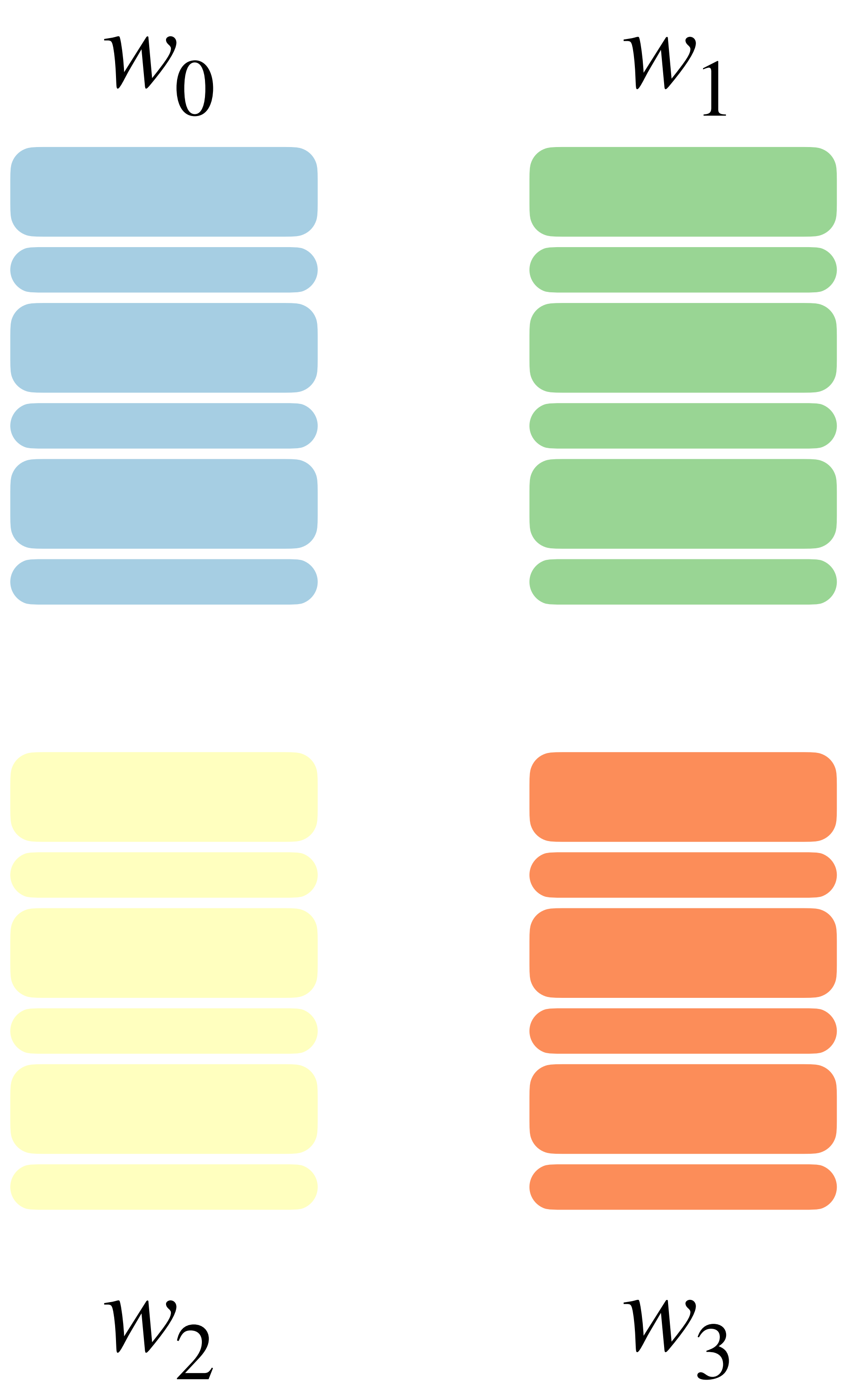
$$y = \frac{1}{n}\sum_{i=0}^{n-1} f(w_i, x)$$

Common approaches for boosting performance at the last stage of pre-training (with slightly different data mixes) or for specialization (by dividing public data into several domain categories)

Li et al. 2022. "Branch-Train-Merge: Embarrassingly Parallel Training of Expert Language Models"
Gururangan et al. 2023. "Scaling Expert Language Models with Unsupervised Domain Discovery"

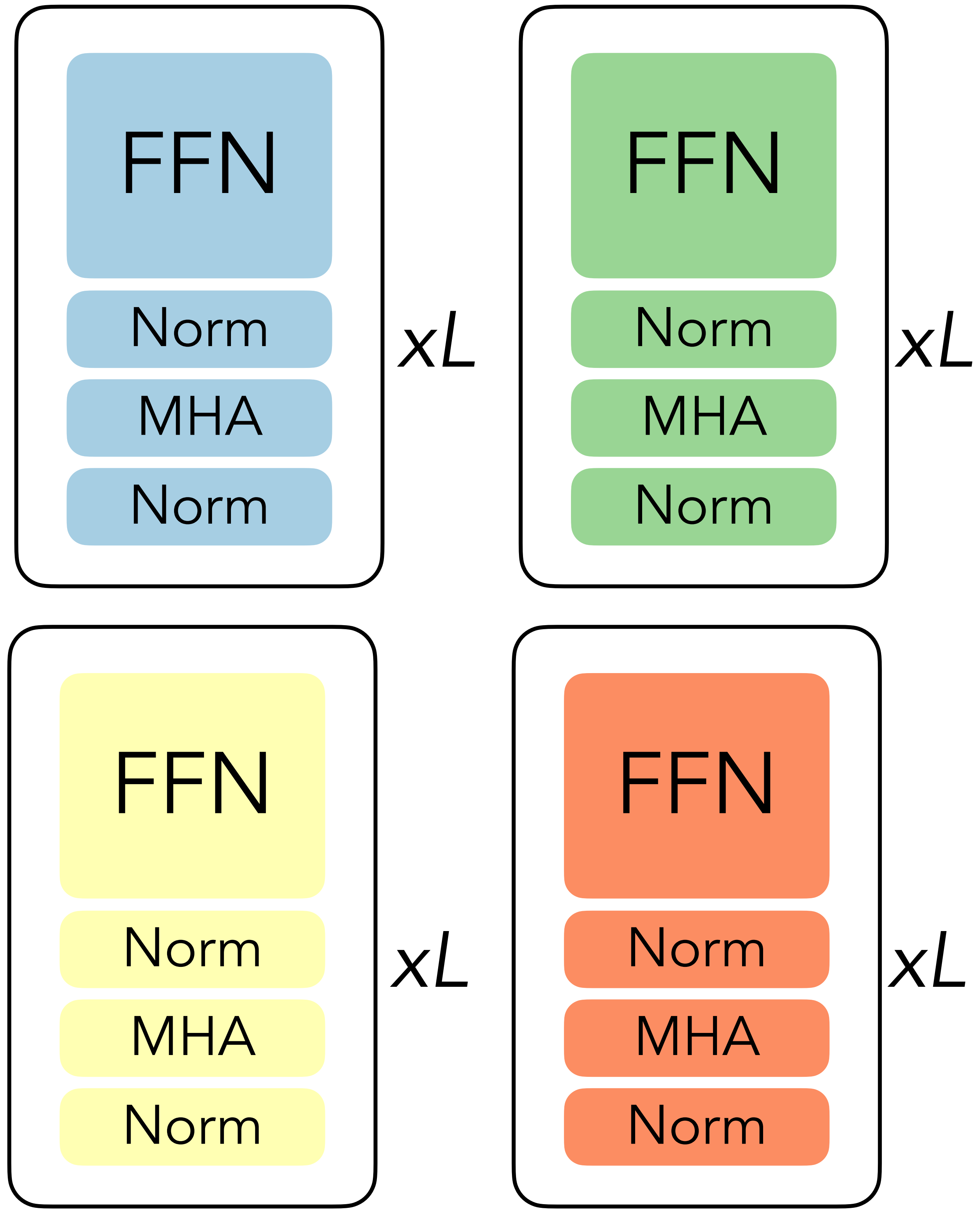
Existing approach 2: MoEification

How to merge n different LMs, $w_0, w_1, \cdots, w_{n-1}$, to boost performance?



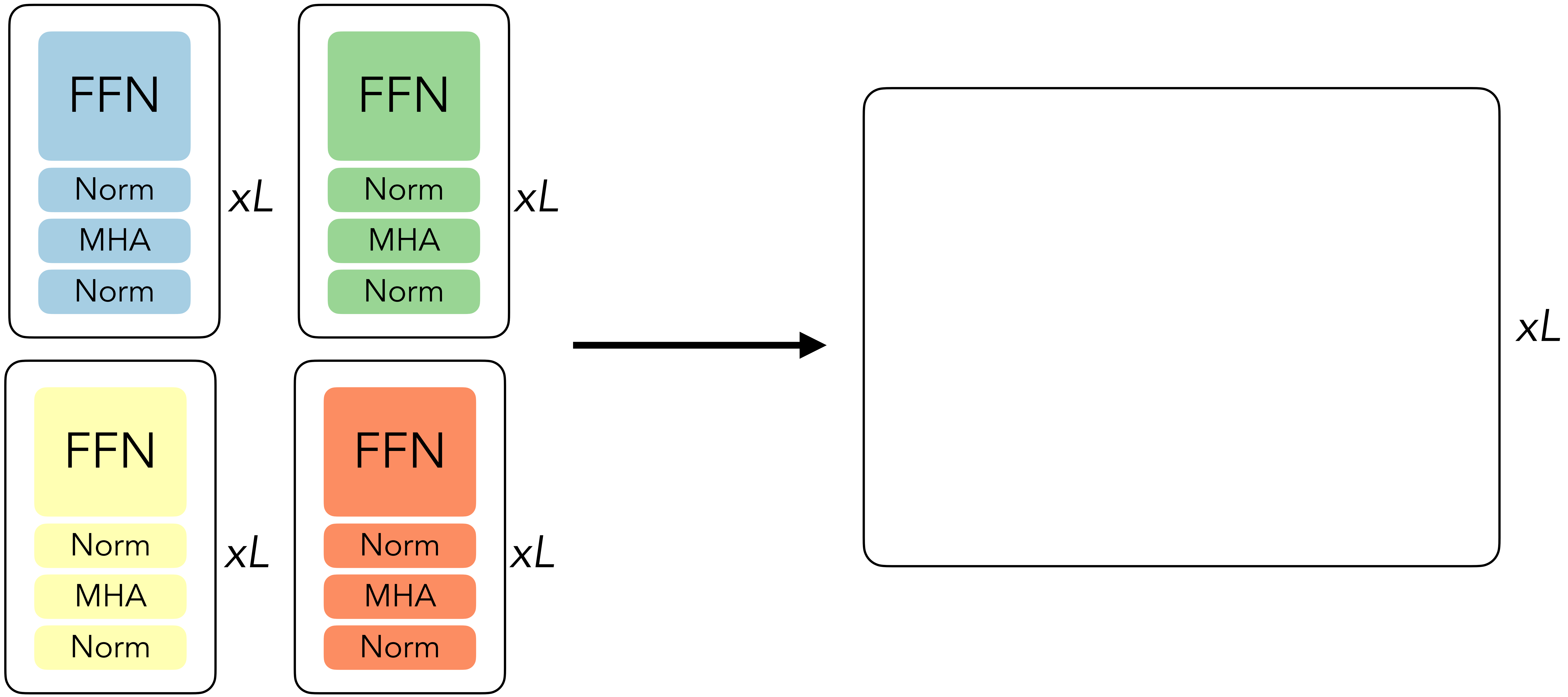
Sukhbaatar et al. 2024. "Branch-Train-MiX: Mixing Expert LLMs into a Mixture-of-Experts LLM"
Gritsch et al. 2024. "Nexus: Specialization meets Adaptability for Efficiently Training Mixture of Experts"
Schafhalter et al. 2024. "Scalable Multi-Domain Adaptation of Language Models using Modular Experts"

Existing approach 2: MoEification



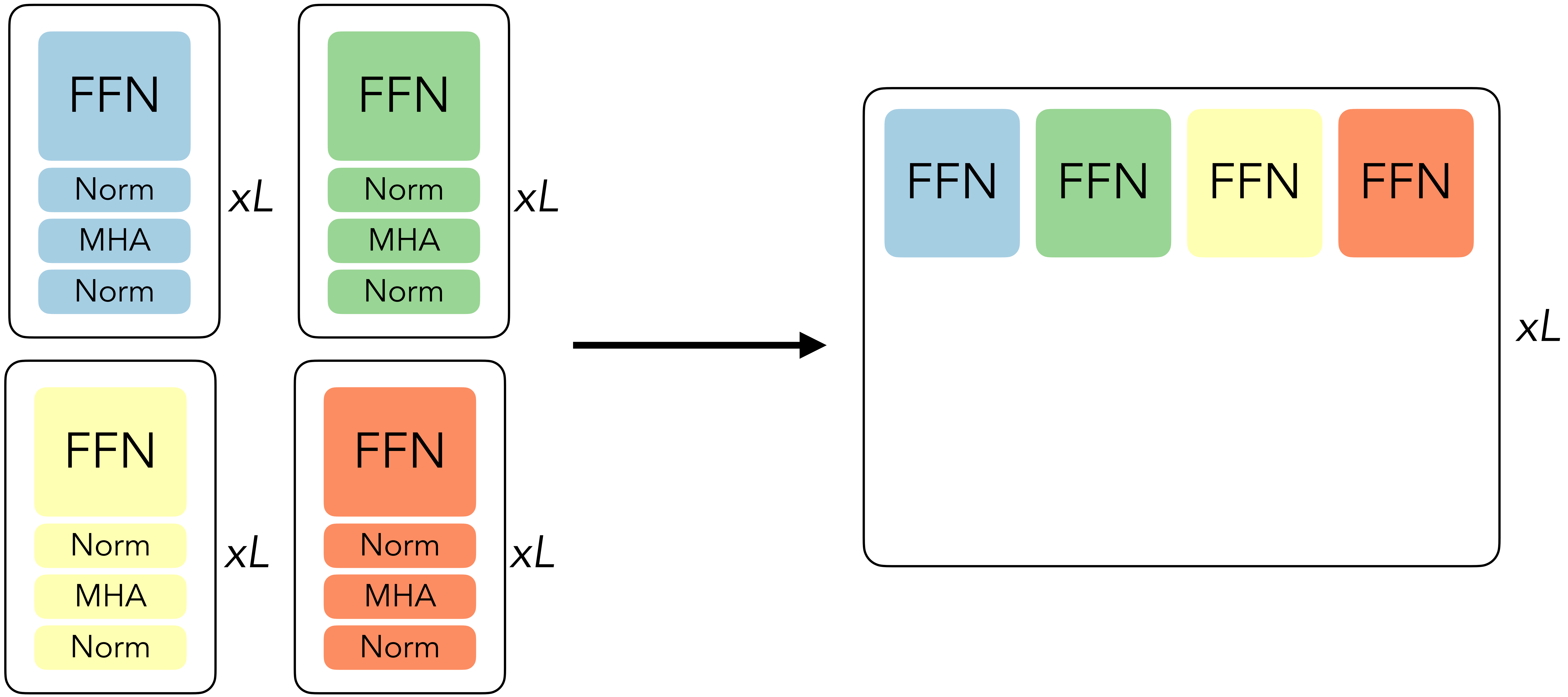
Sukhbaatar et al. 2024. "Branch-Train-MiX: Mixing Expert LLMs into a Mixture-of-Experts LLM"
Gritsch et al. 2024. "Nexus: Specialization meets Adaptability for Efficiently Training Mixture of Experts"
Schafhalter et al. 2024. "Scalable Multi-Domain Adaptation of Language Models using Modular Experts"

Existing approach 2: MoEification



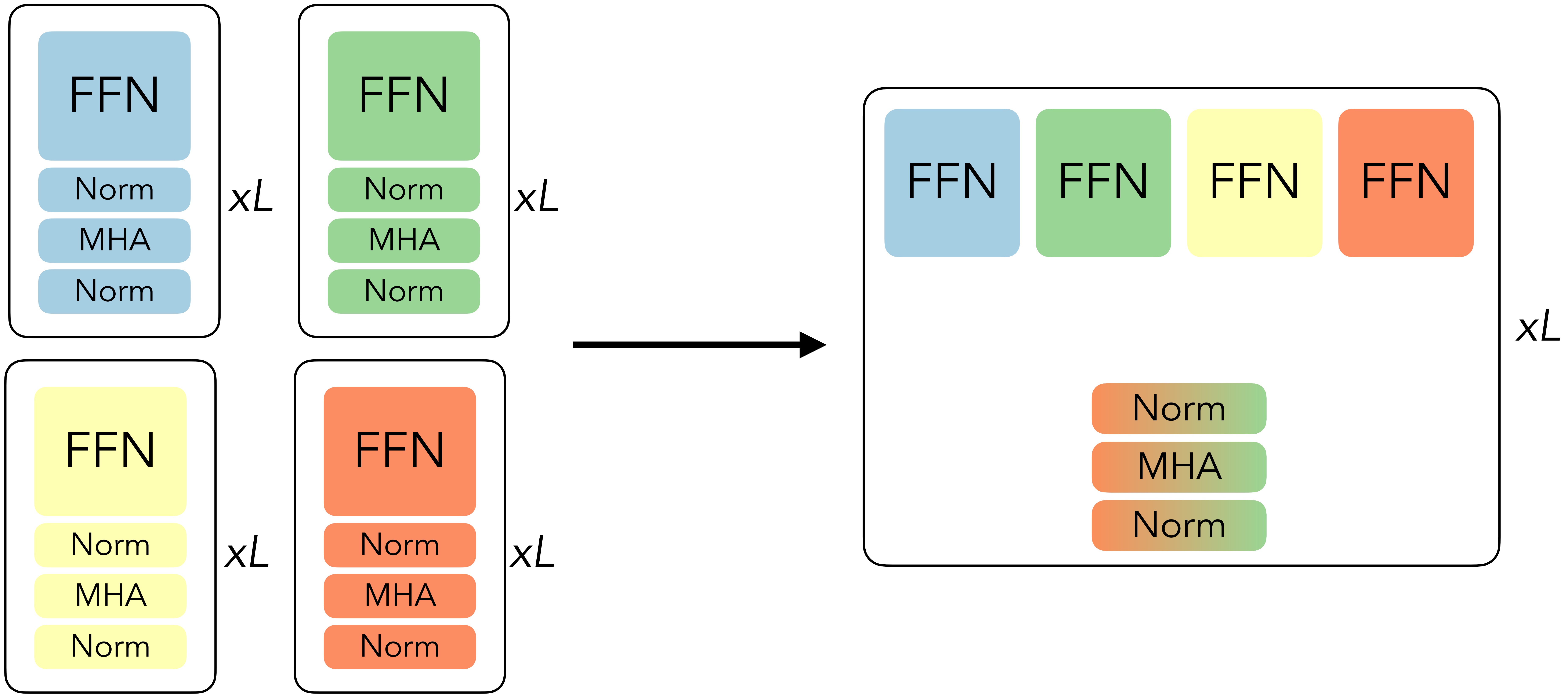
Sukhbaatar et al. 2024. "Branch-Train-MiX: Mixing Expert LLMs into a Mixture-of-Experts LLM"
Gritsch et al. 2024. "Nexus: Specialization meets Adaptability for Efficiently Training Mixture of Experts"
Schafhalter et al. 2024. "Scalable Multi-Domain Adaptation of Language Models using Modular Experts"

Existing approach 2: MoEification



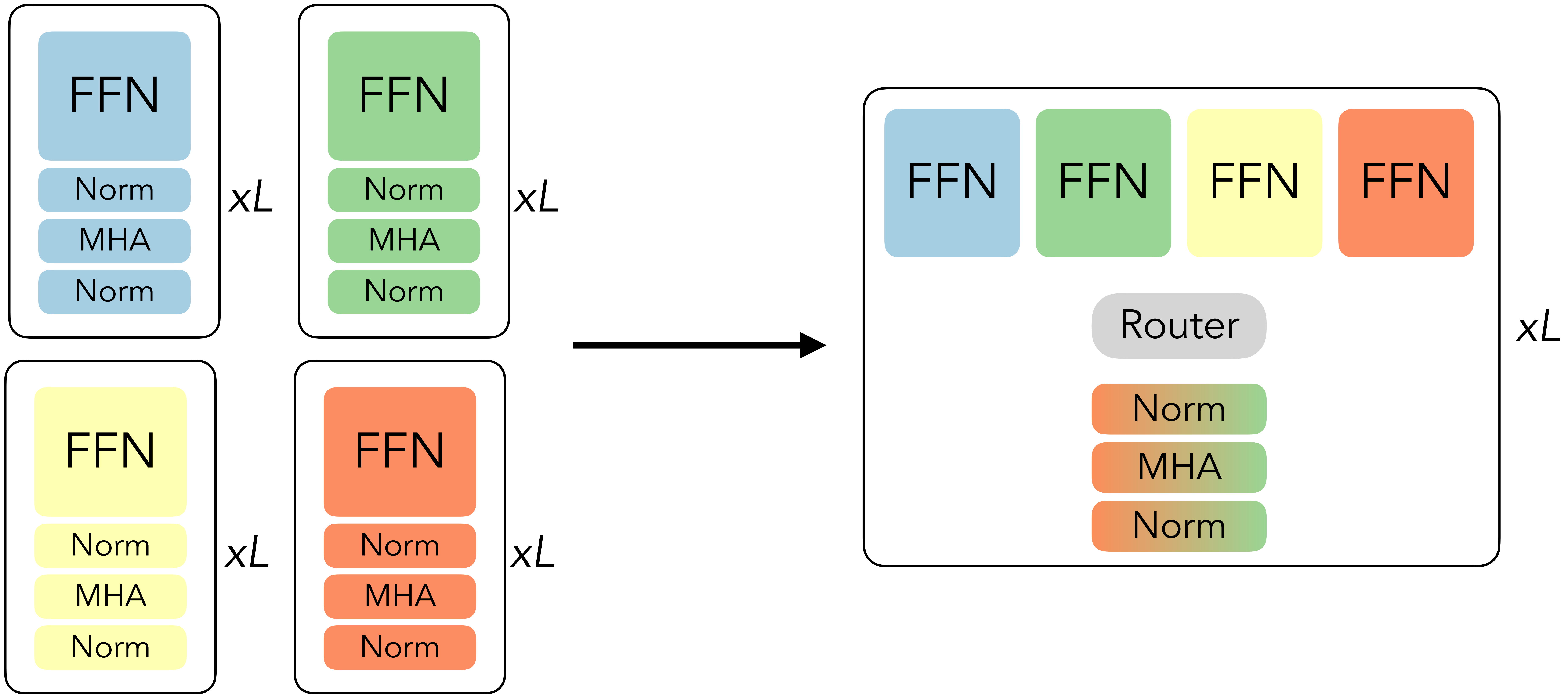
Sukhbaatar et al. 2024. ”Branch-Train-MiX: Mixing Expert LLMs into a Mixture-of-Experts LLM”
Gritsch et al. 2024. "Nexus: Specialization meets Adaptability for Efficiently Training Mixture of Experts”
Schafhalter et al. 2024. "Scalable Multi-Domain Adaptation of Language Models using Modular Experts"

Existing approach 2: MoEification



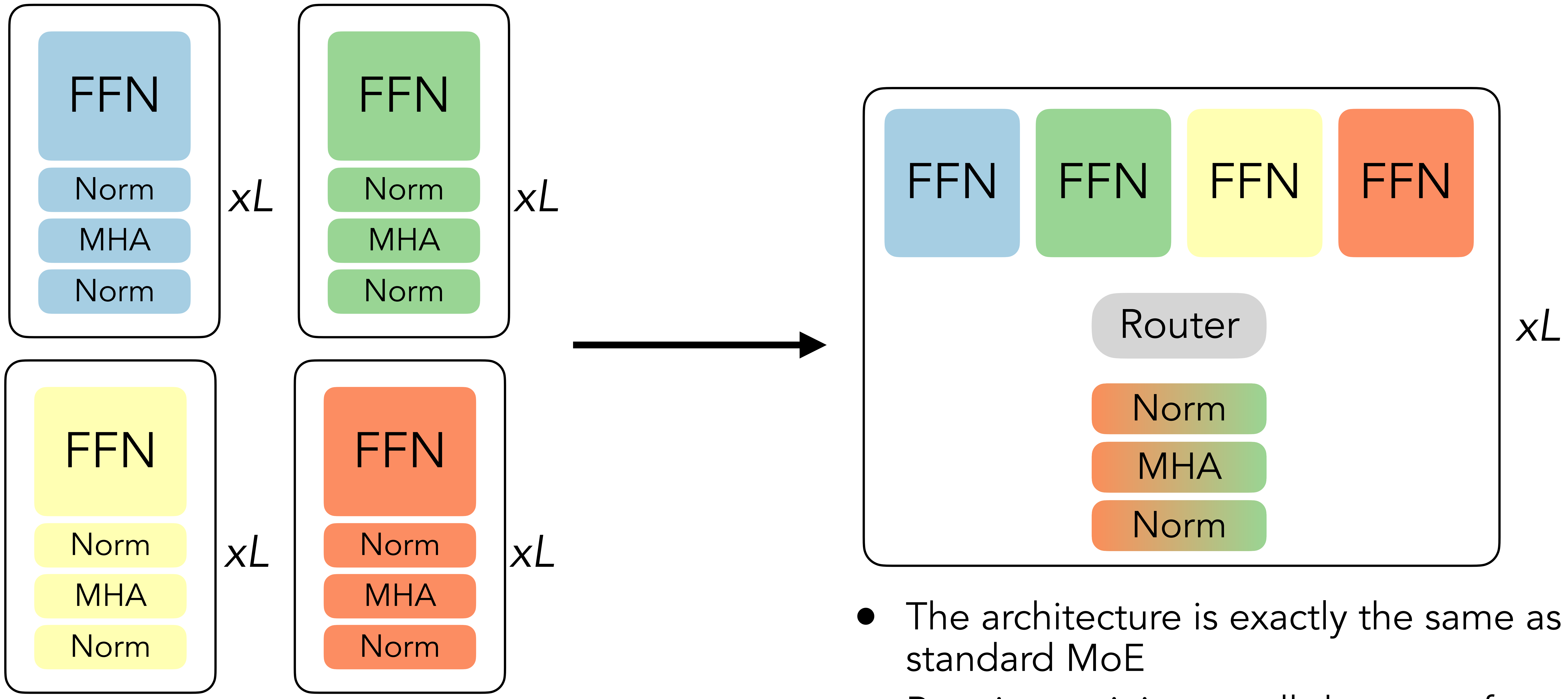
Sukhbaatar et al. 2024. "Branch-Train-MiX: Mixing Expert LLMs into a Mixture-of-Experts LLM"
Gritsch et al. 2024. "Nexus: Specialization meets Adaptability for Efficiently Training Mixture of Experts"
Schafhalter et al. 2024. "Scalable Multi-Domain Adaptation of Language Models using Modular Experts"

Existing approach 2: MoEification



Sukhbaatar et al. 2024. "Branch-Train-MiX: Mixing Expert LLMs into a Mixture-of-Experts LLM"
Gritsch et al. 2024. "Nexus: Specialization meets Adaptability for Efficiently Training Mixture of Experts"
Schafhalter et al. 2024. "Scalable Multi-Domain Adaptation of Language Models using Modular Experts"

Existing approach 2: MoEification



- The architecture is exactly the same as the standard MoE
- Requires training on all datasets after merging

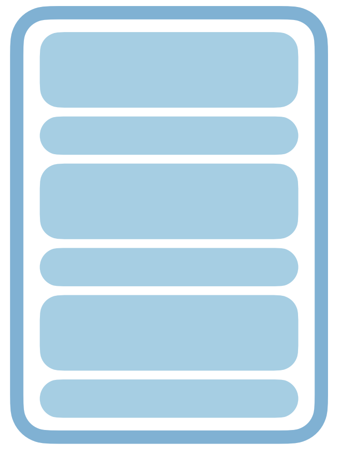
Sukhbaatar et al. 2024. "Branch-Train-MiX: Mixing Expert LLMs into a Mixture-of-Experts LLM"
Gritsch et al. 2024. "Nexus: Specialization meets Adaptability for Efficiently Training Mixture of Experts"
Schafhalter et al. 2024. "Scalable Multi-Domain Adaptation of Language Models using Modular Experts"

Applying to our setup

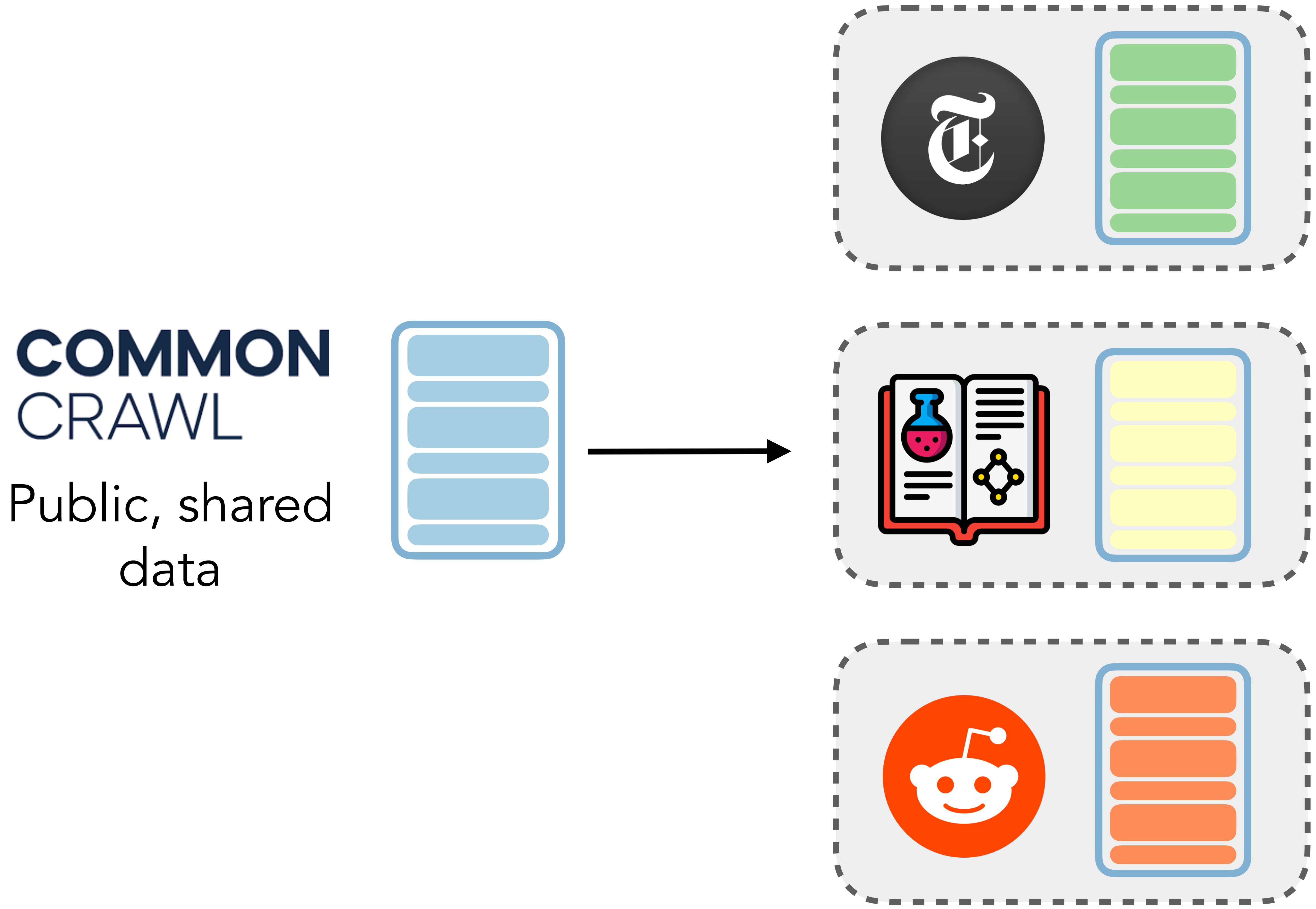
Applying to our setup

COMMON
CRAWL

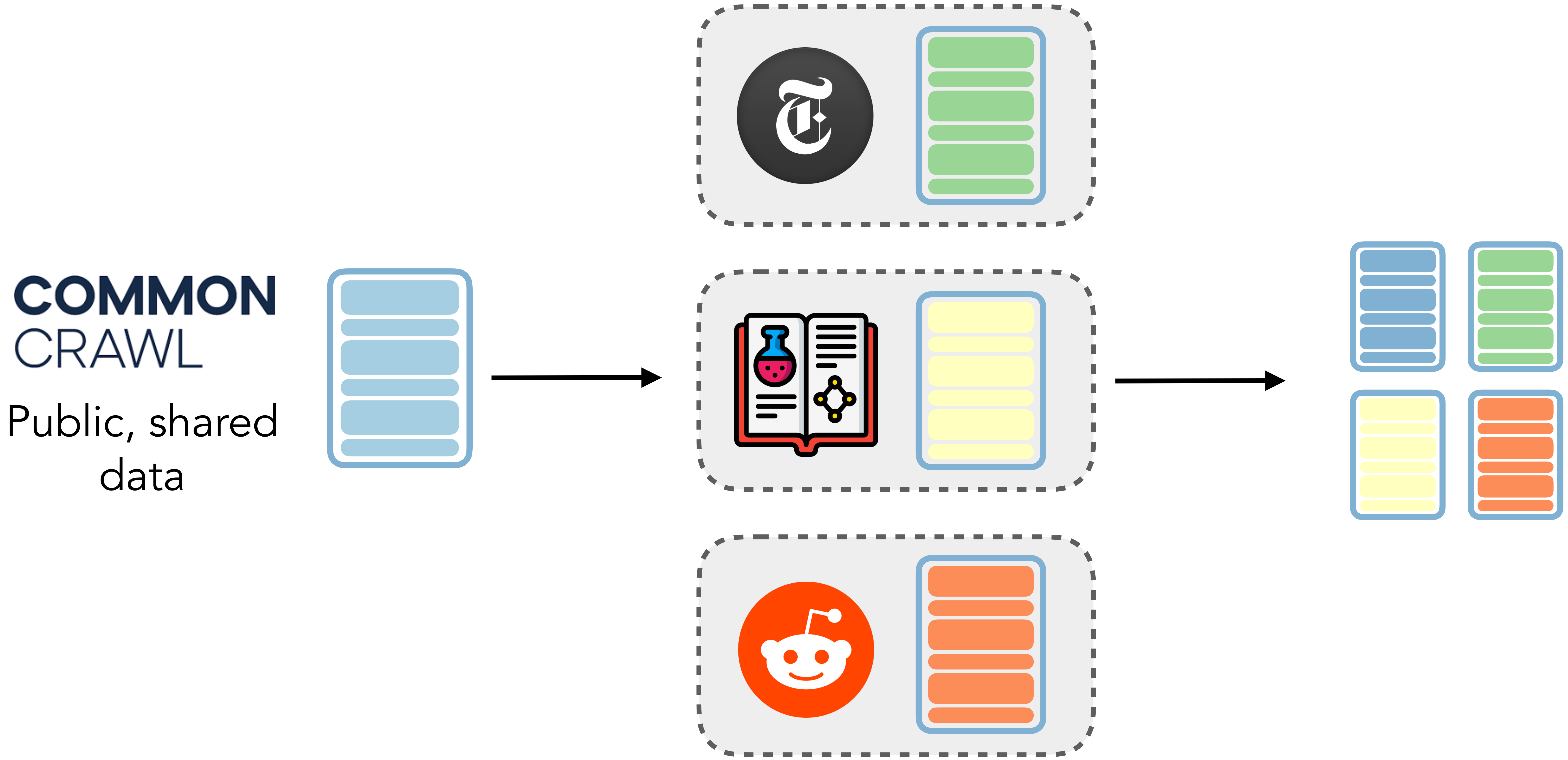
Public, shared data



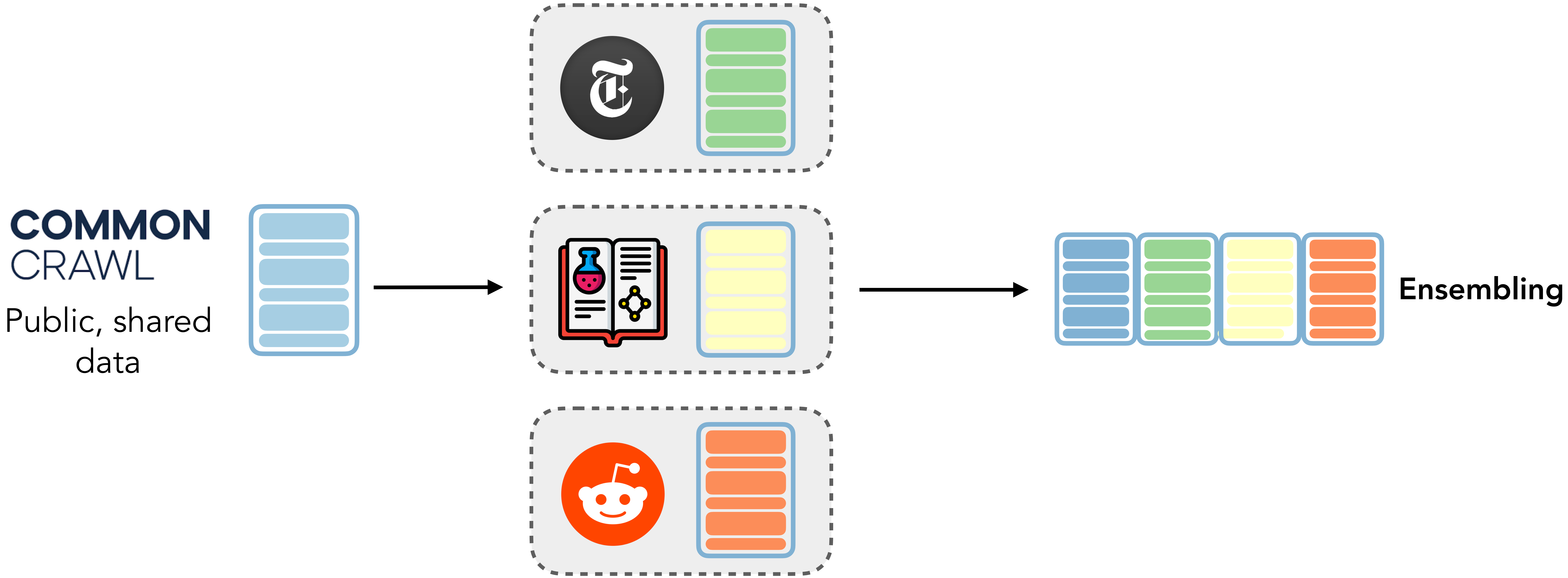
Applying to our setup



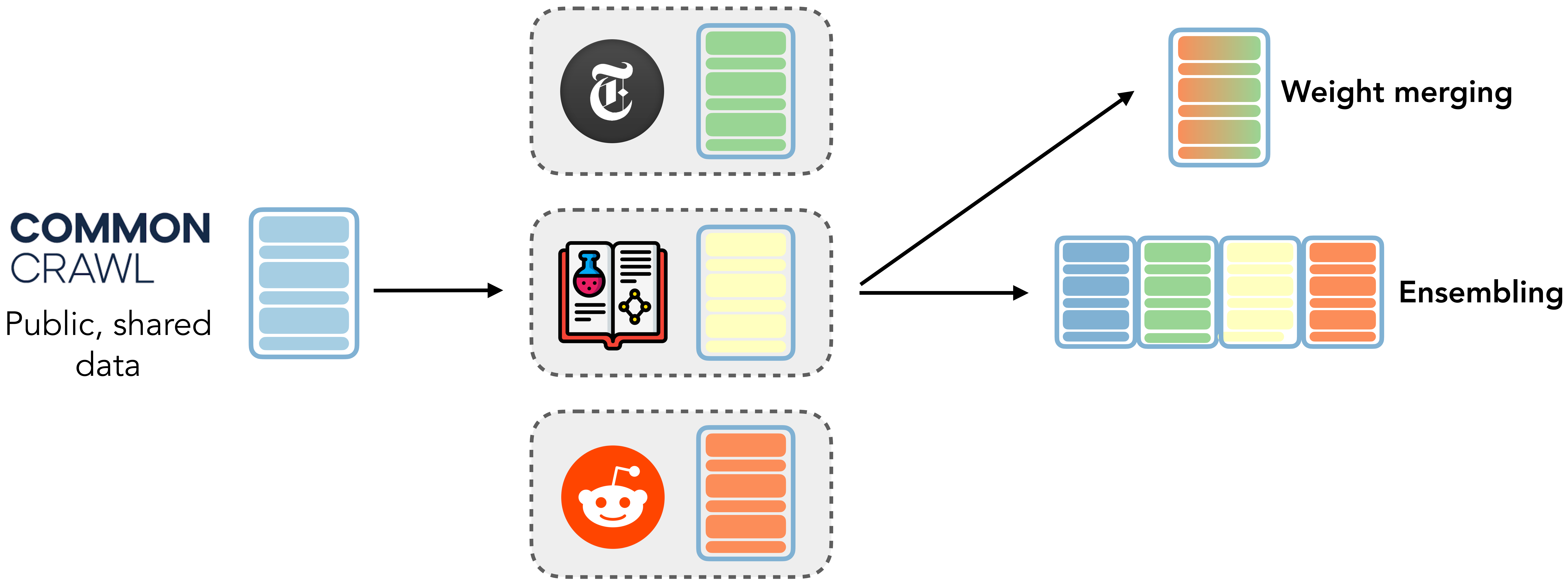
Applying to our setup



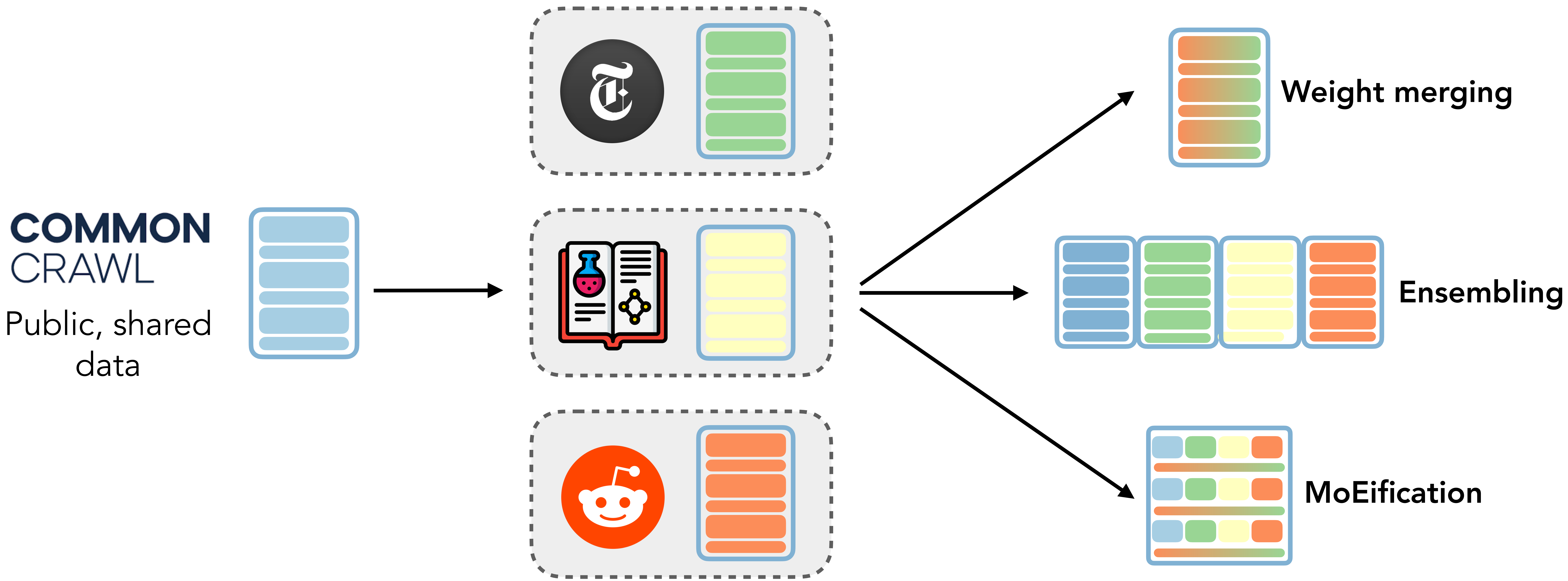
Applying to our setup



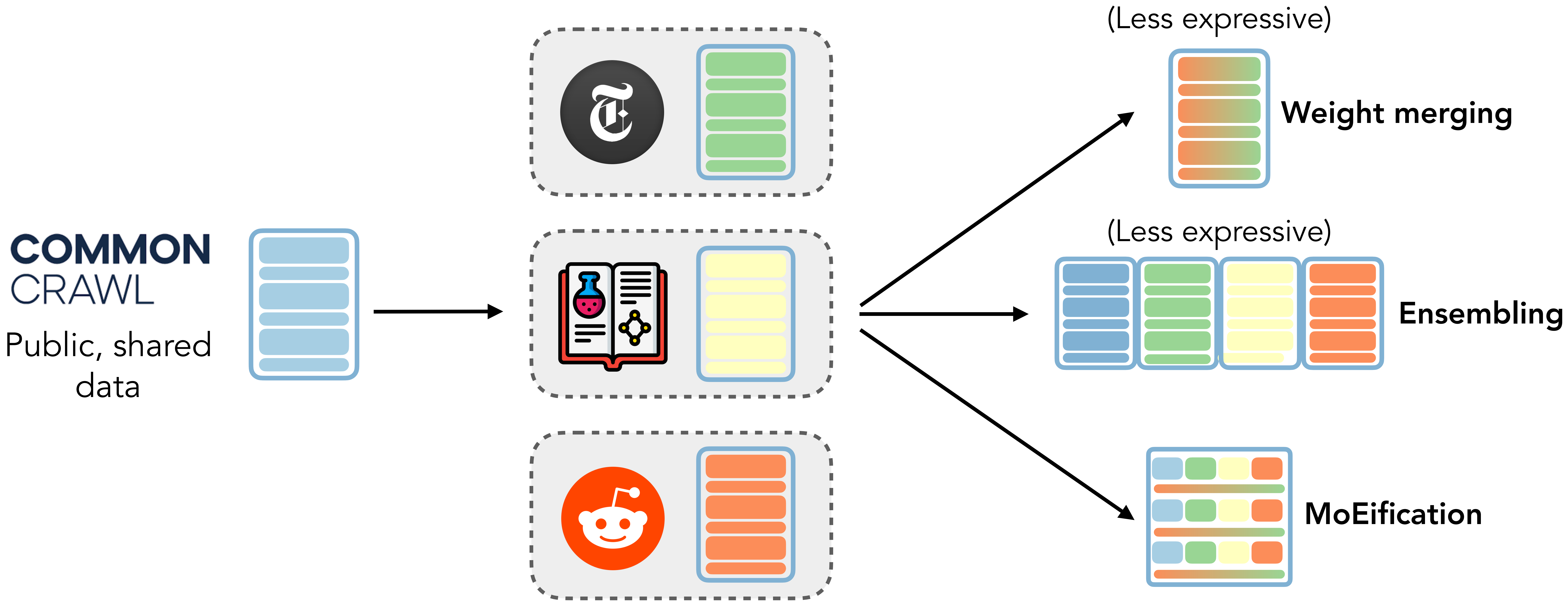
Applying to our setup



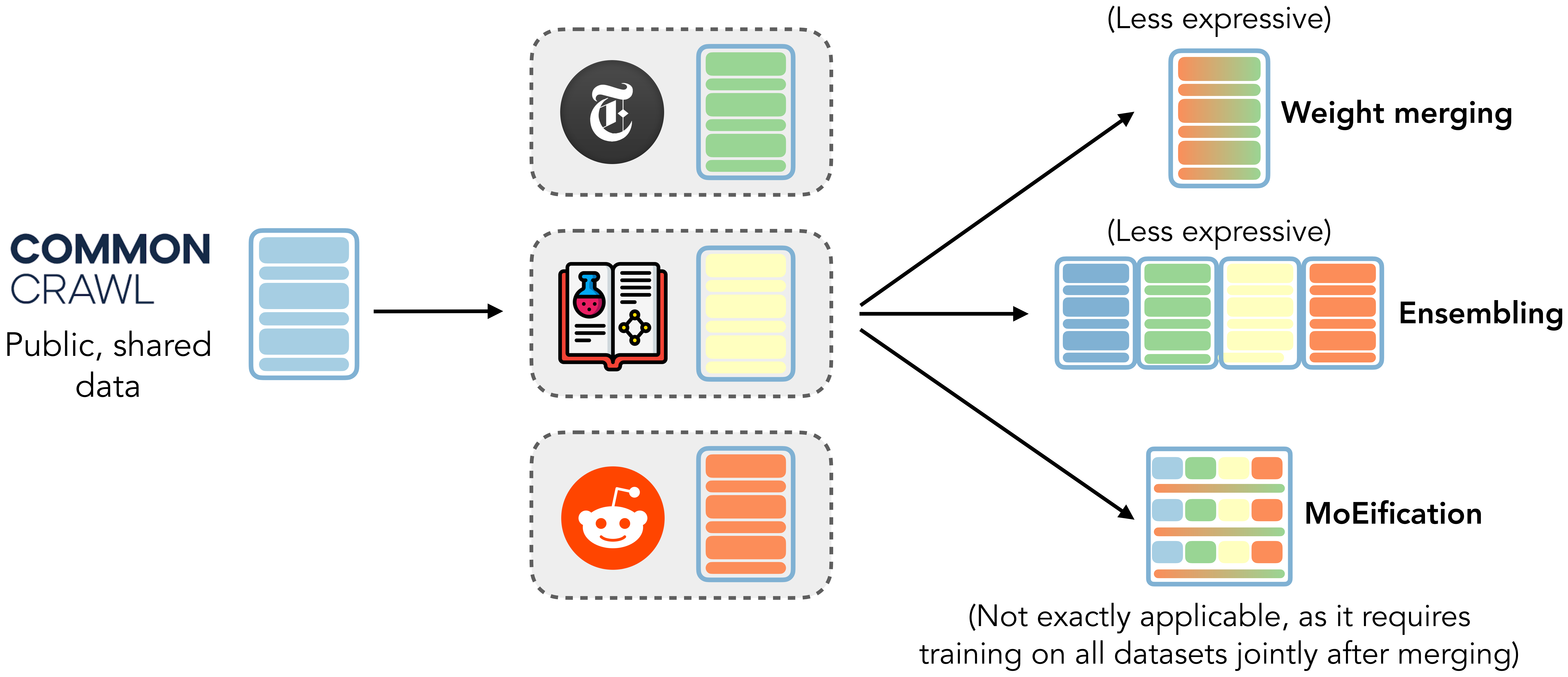
Applying to our setup



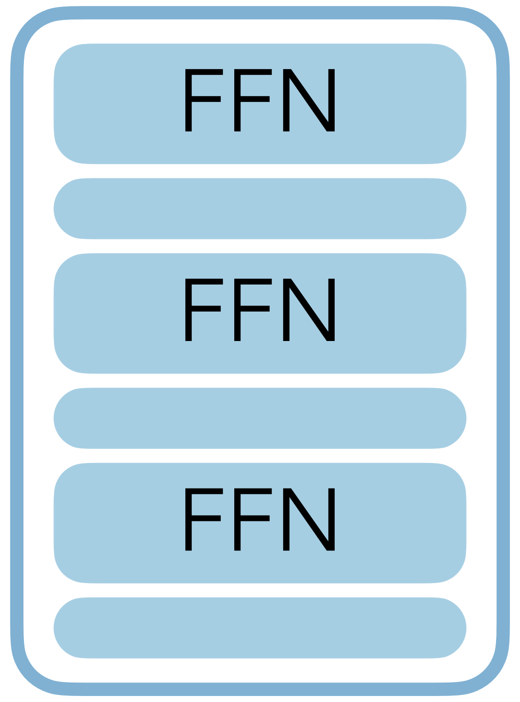
Applying to our setup



Applying to our setup

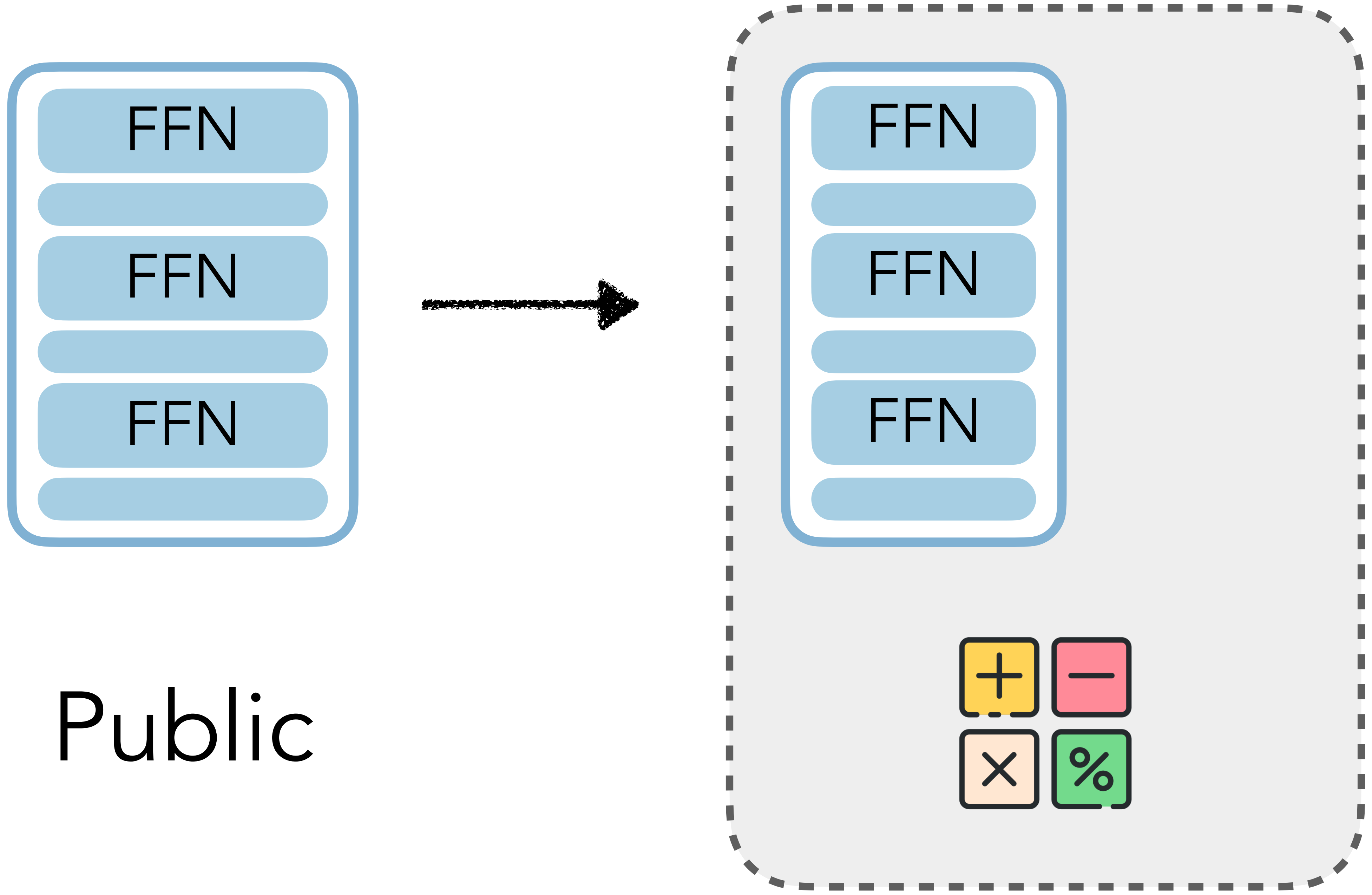


Idea 1: MoE-aware siloed training

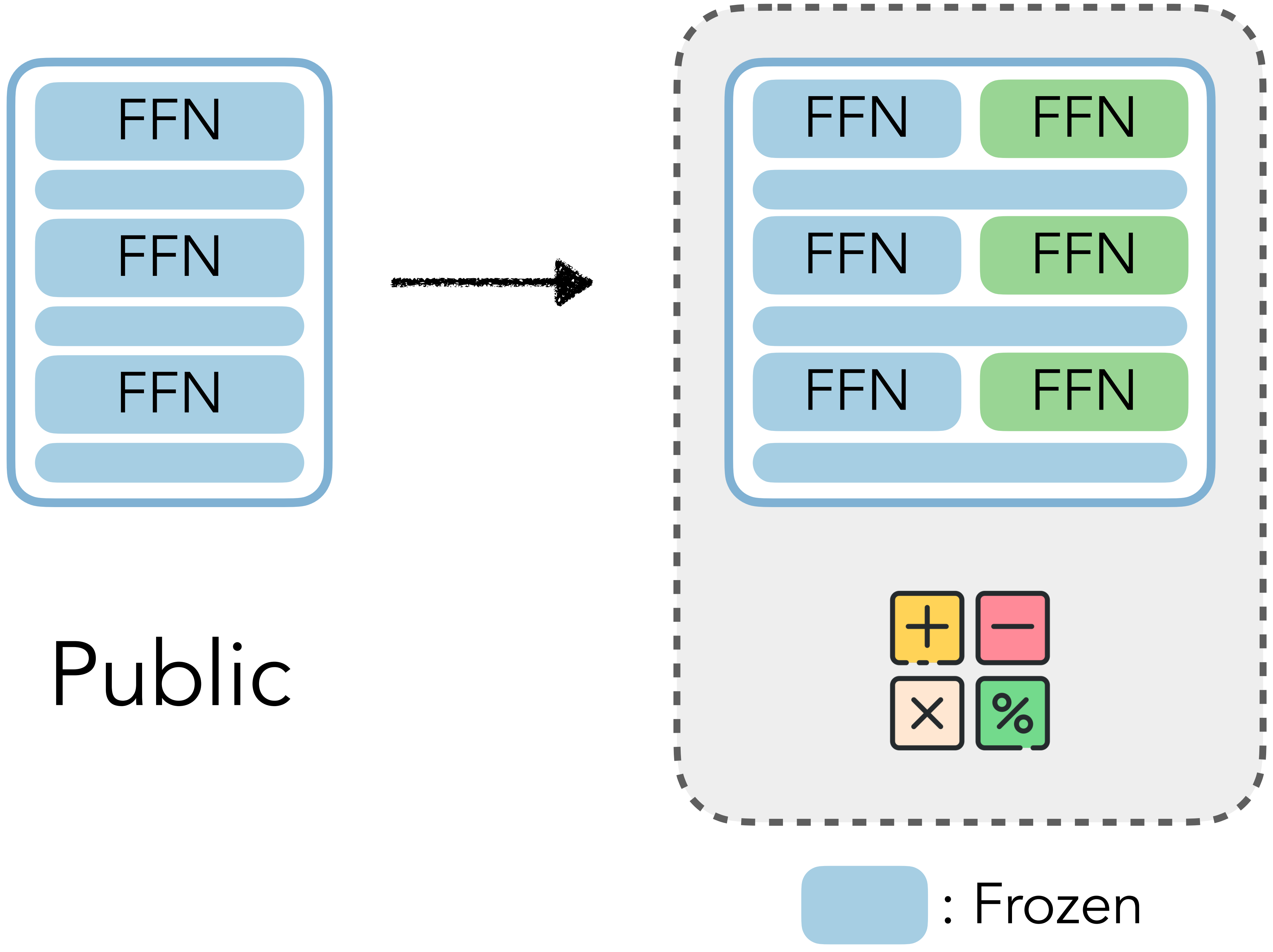


Public

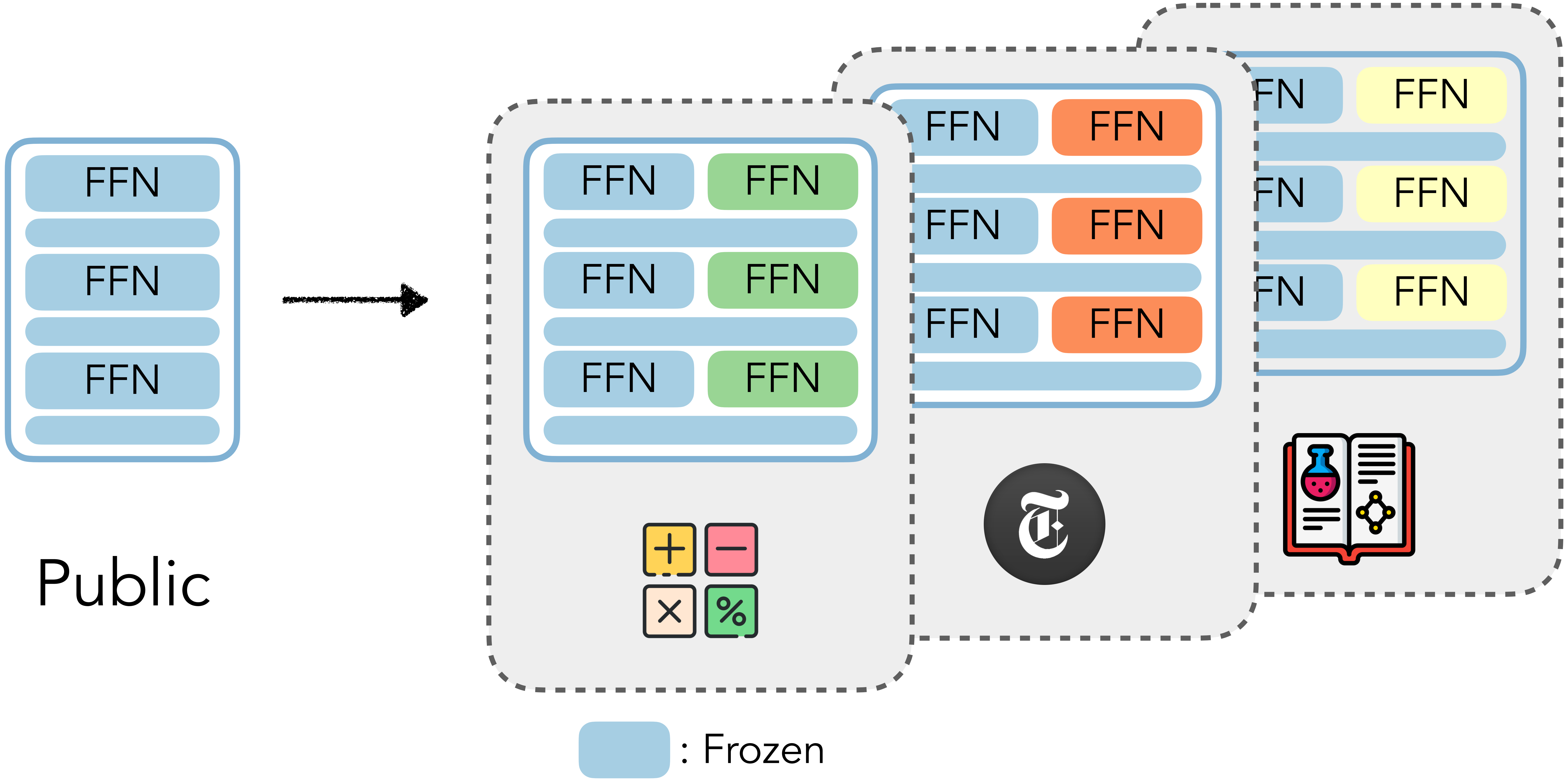
Idea 1: MoE-aware siloed training



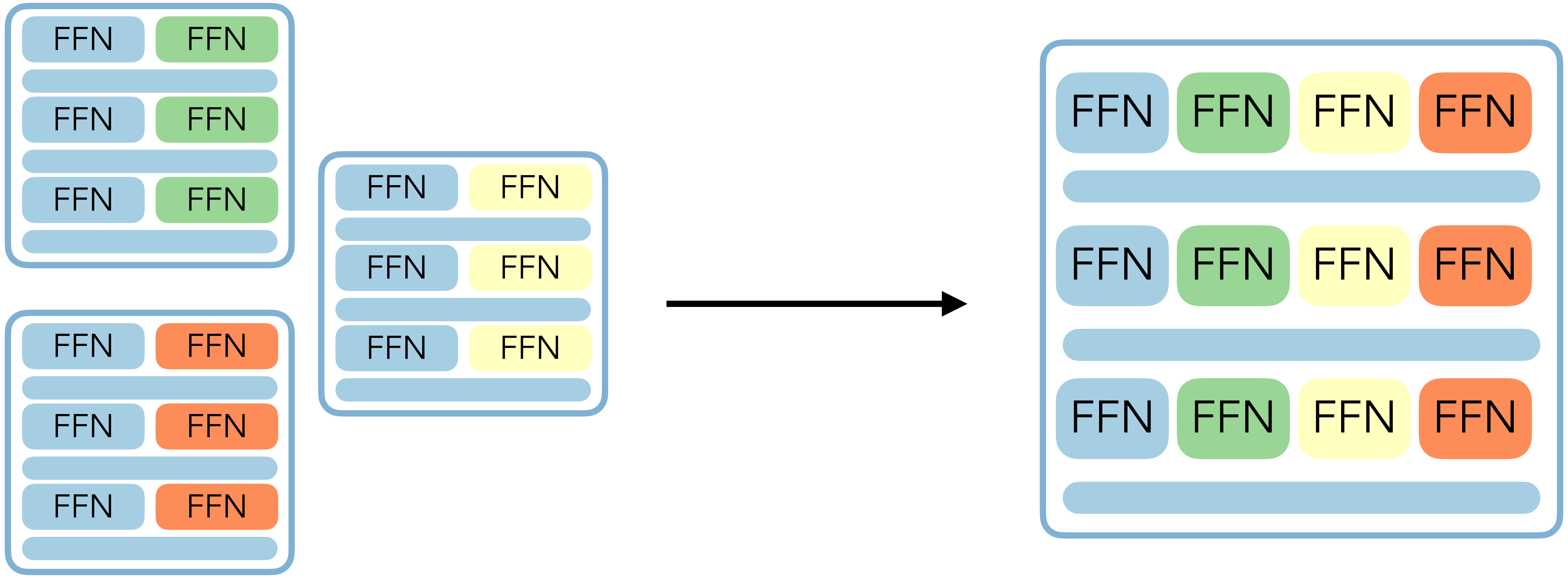
Idea 1: MoE-aware siloed training



Idea 1: MoE-aware siloed training



Idea 1: MoE-aware siloed training



- Works OK even without router training after merging
- However, errors still come from suboptimal router decisions

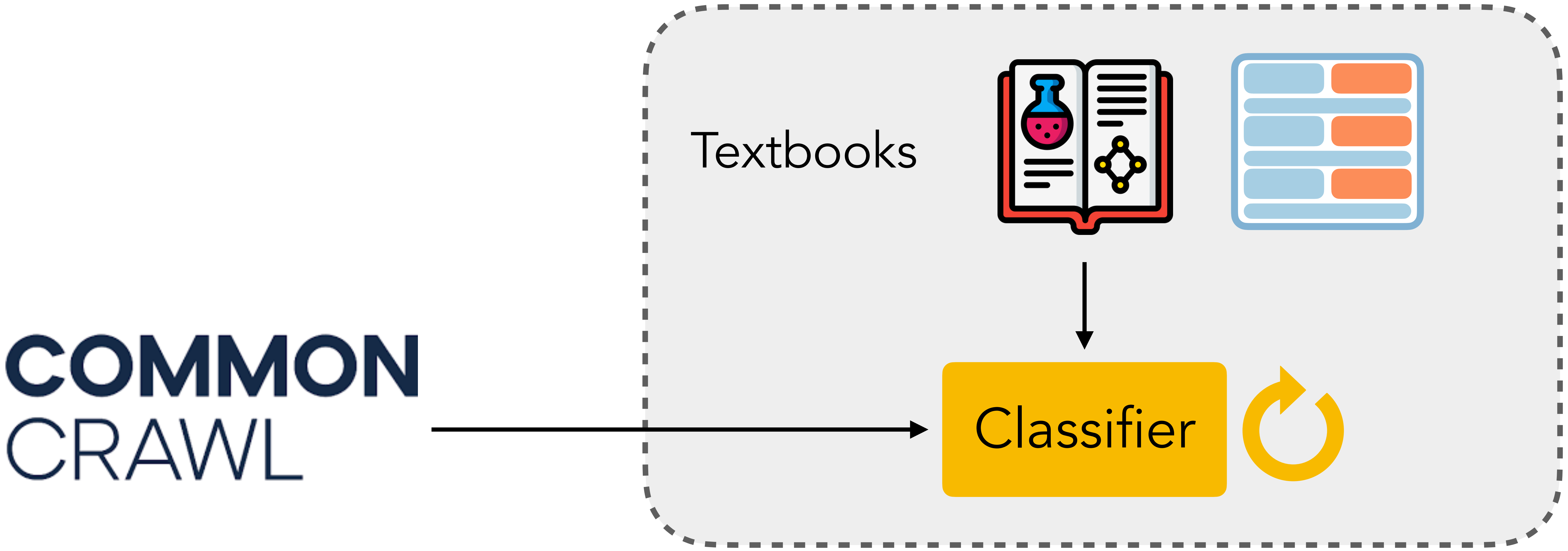
Idea 2: Obtain proxy data to siloed data

Q: Can we find a subset of public data that looks like each siloed data?

e.g., a subset of Common Crawl that looks like textbooks?

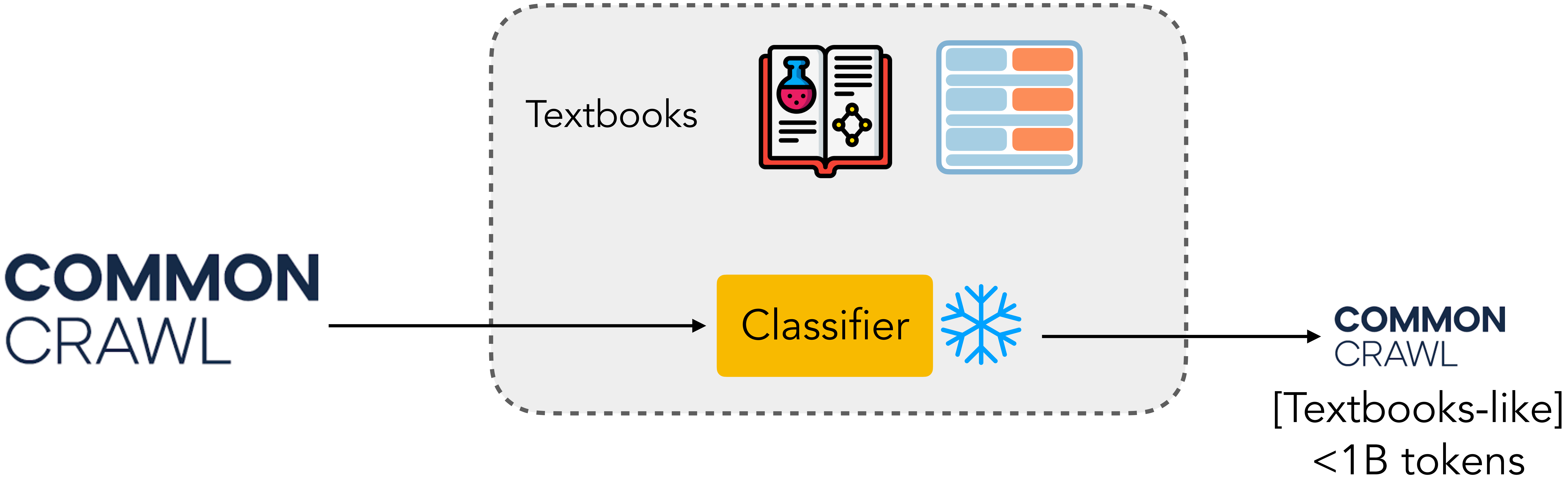
Idea 2: Obtain proxy data to siloed data

Q: Can we find a subset of public data that looks like each siloed data?
e.g., a subset of Common Crawl that looks like textbooks?



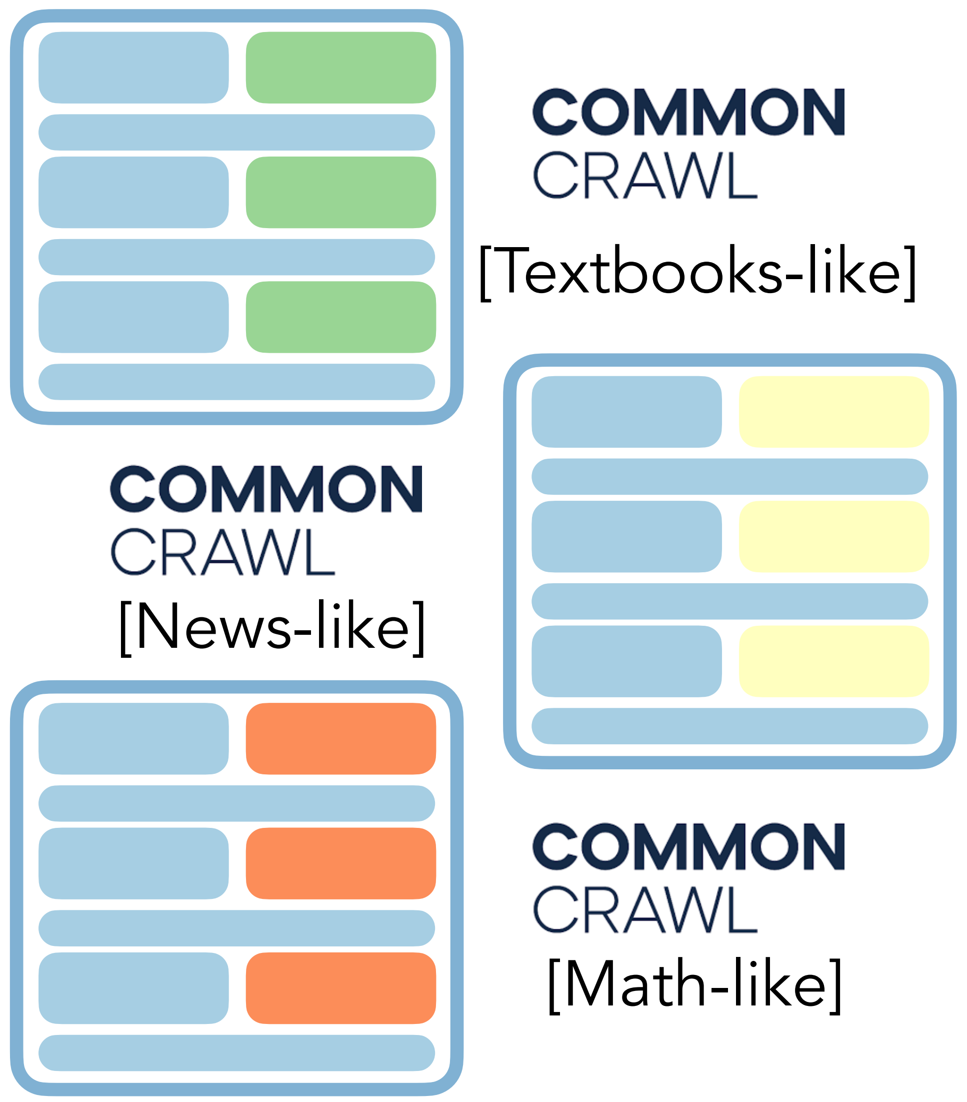
Idea 2: Obtain proxy data to siloed data

Q: Can we find a subset of public data that looks like each siloed data?
e.g., a subset of Common Crawl that looks like textbooks?



Idea 2: Obtain proxy data to siloed data

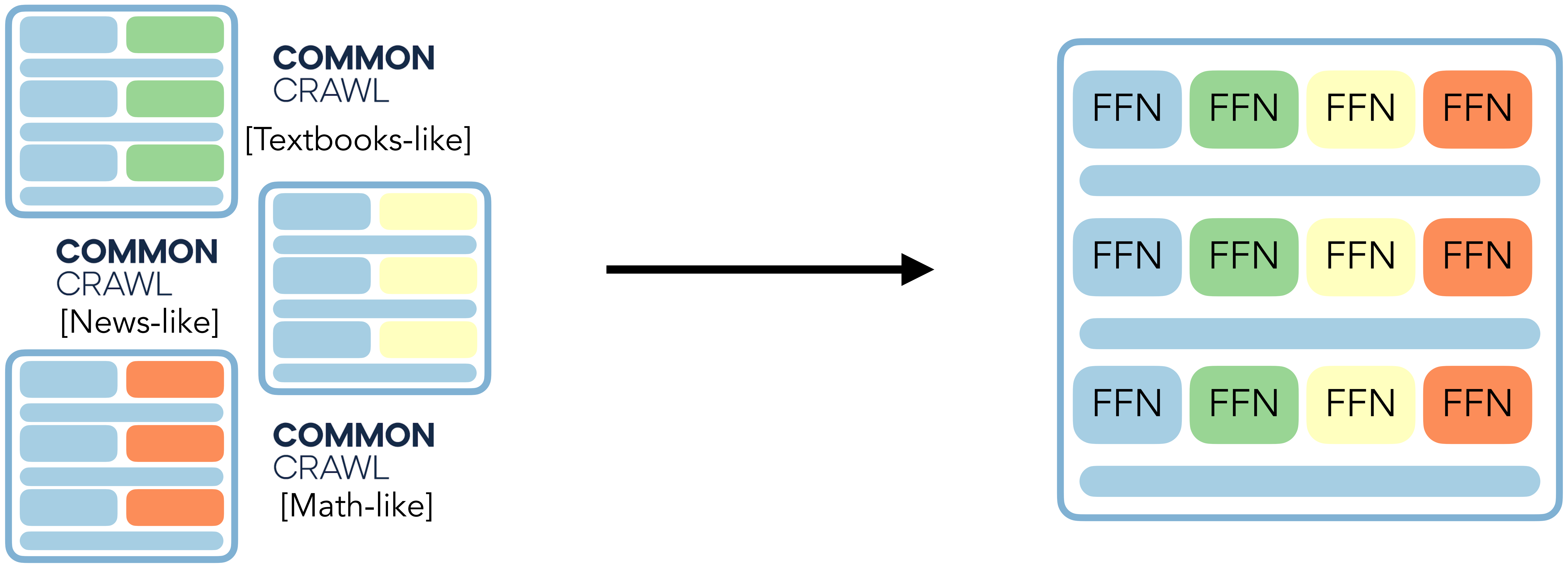
Q: Can we find a subset of public data that looks like each siloed data?
e.g., a subset of Common Crawl that looks like textbooks?



Idea 2: Obtain proxy data to siloed data

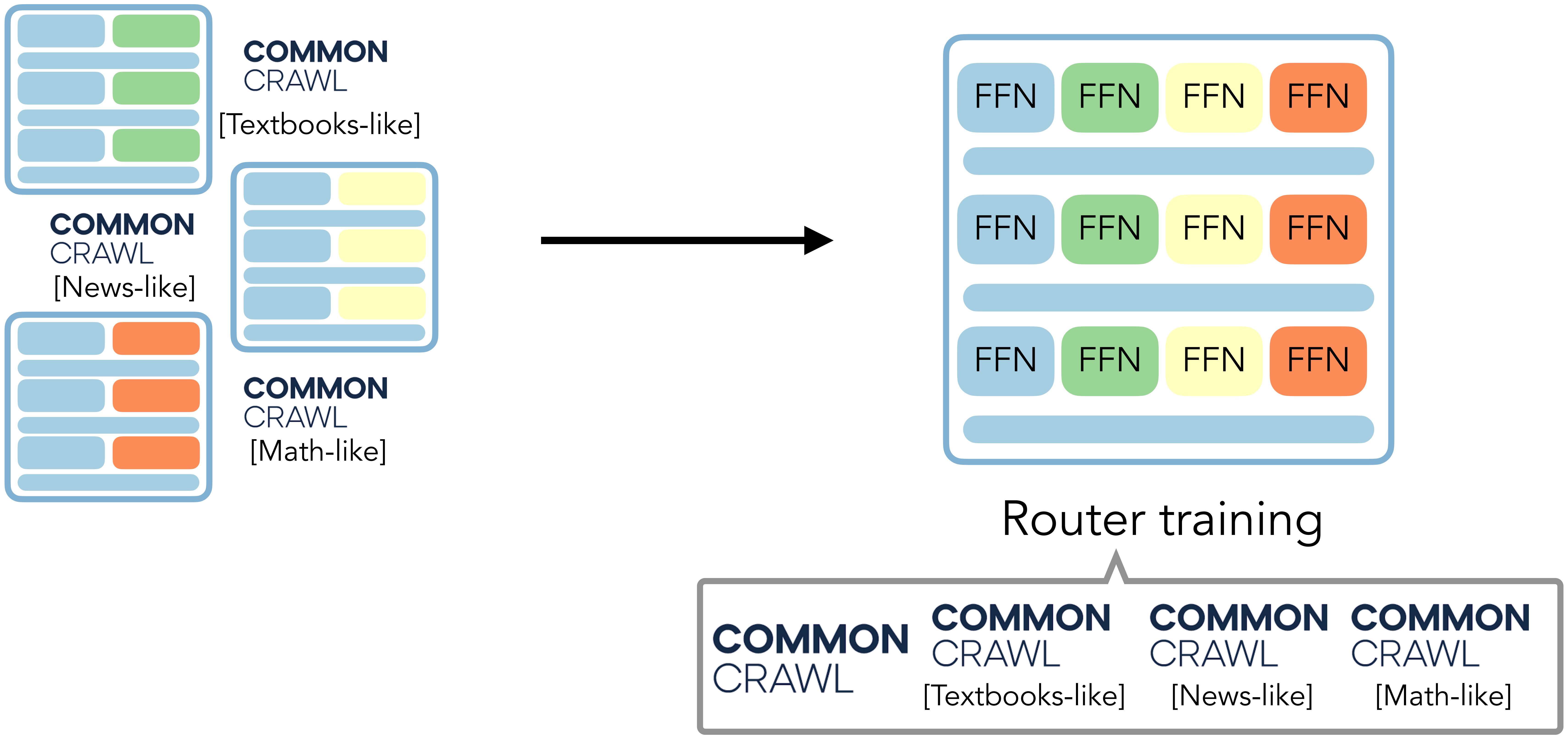
Q: Can we find a subset of public data that looks like each siloed data?

e.g., a subset of Common Crawl that looks like textbooks?



Idea 2: Obtain proxy data to siloed data

Q: Can we find a subset of public data that looks like each siloed data?
e.g., a subset of Common Crawl that looks like textbooks?



Experiments

Experimental setup

- **7B parameters**
- Hidden dimension: 4096
- Attention heads: 32
- Batch size: 1024
- Sequence length: 4096
- Peak learning rate: 9e-4
- LR warmup 2000 steps
- Learning rate schedule: Cosine decay until 5T tokens, truncated at 1B, then annealed

Pre-training on shared data for **1T tokens**

↓

Contunue-pre-training on each siloed data for **50B tokens**

↓

(Optional) After merging, router training for **5B tokens**

Task	Public model
Core 9	68.4
MMLU Biology	63.0
MMLU CS	52.2
MMLU Economics	45.7
MMLU Engineering	46.0
MMLU Health	59.5
MMLU Physics	45.8
MMLU Politics	73.0
MMLU Pro Biology	56.0
MMLU Pro Engineering	18.0
AGI Eval	39.0
Big Bench Hard	35.6
GSM8K	12.0
MATH	4.3
Four coding tasks	1.0
AVERAGE	41.3

*ARC easy, ARC challenge, BoolQ, CSQA, Hellaswag, OpenbookQA, PIQA, SIQA, WG

*HumanEval, HumanEval Plus, MBPP, MBPP Plus

Task	Public model	Textbooks Expert
Core 9	68.4	63.0
MMLU Biology	63.0	**66.5**
MMLU CS	52.2	**55.0**
MMLU Economics	45.7	**49.3**
MMLU Engineering	46.0	**51.0**
MMLU Health	59.5	59.8
MMLU Physics	45.8	**49.2**
MMLU Politics	73.0	67.5
MMLU Pro Biology	56.0	54.0
MMLU Pro Engineering	18.0	17.0
AGI Eval	39.0	39.6
Big Bench Hard	35.6	**40.0**
GSM8K	12.0	**20.0**
MATH	4.3	6.3
Four coding tasks	1.0	4.3
AVERAGE	41.3	42.8

Task	Public model	Textbooks Expert
Core 9	68.4	63.0
MMLU Biology	63.0	**66.5**
MMLU CS	52.2	**55.0**
MMLU Economics	45.7	**49.3**
MMLU Engineering	46.0	**51.0**
MMLU Health	59.5	59.8
MMLU Physics	45.8	**49.2**
MMLU Politics	73.0	67.5
MMLU Pro Biology	56.0	54.0
MMLU Pro Engineering	18.0	17.0
AGI Eval	39.0	39.6
Big Bench Hard	35.6	**40.0**
GSM8K	12.0	**20.0**
MATH	4.3	6.3
Four coding tasks	1.0	4.3
AVERAGE	41.3	42.8

Task	Public model	Textbooks Expert	Math Expert
Core 9	68.4	63.0	63.8
MMLU Biology	63.0	66.5	52.5
MMLU CS	52.2	55.0	53.7
MMLU Economics	45.7	49.3	46.0
MMLU Engineering	46.0	51.0	**62.0**
MMLU Health	59.5	59.8	44.3
MMLU Physics	45.8	49.2	47.7
MMLU Politics	73.0	67.5	62.5
MMLU Pro Biology	56.0	54.0	45.0
MMLU Pro Engineering	18.0	17.0	15.0
AGI Eval	39.0	39.6	40.7
Big Bench Hard	35.6	40.0	**45.4**
GSM8K	12.0	20.0	**68.0**
MATH	4.3	6.3	**32.9**
Four coding tasks	1.0	4.3	**18.1**
AVERAGE	41.3	42.8	46.5

Task	Public model	Textbooks Expert	Math Expert
Core 9	68.4	63.0	63.8
MMLU Biology	63.0	66.5	52.5
MMLU CS	52.2	55.0	53.7
MMLU Economics	45.7	49.3	46.0
MMLU Engineering	46.0	51.0	**62.0**
MMLU Health	59.5	59.8	44.3
MMLU Physics	45.8	49.2	47.7
MMLU Politics	73.0	67.5	62.5
MMLU Pro Biology	56.0	54.0	45.0
MMLU Pro Engineering	18.0	17.0	15.0
AGI Eval	39.0	39.6	40.7
Big Bench Hard	35.6	40.0	**45.4**
GSM8K	12.0	20.0	**68.0**
MATH	4.3	6.3	**32.9**
Four coding tasks	1.0	4.3	**18.1**
AVERAGE	41.3	42.8	46.5

Task	Public model	Textbooks Expert	Math Expert	Code Expert
Core 9	68.4	63.0	63.8	38.7
MMLU Biology	63.0	66.5	52.5	31.0
MMLU CS	52.2	55.0	53.7	36.7
MMLU Economics	45.7	49.3	46.0	26.7
MMLU Engineering	46.0	51.0	62.0	33.0
MMLU Health	59.5	59.8	44.3	30.7
MMLU Physics	45.8	49.2	47.7	26.2
MMLU Politics	73.0	67.5	62.5	32.0
MMLU Pro Biology	56.0	54.0	45.0	34.0
MMLU Pro Engineering	18.0	17.0	15.0	10.0
AGI Eval	39.0	39.6	40.7	29.0
Big Bench Hard	35.6	40.0	45.4	38.2
GSM8K	12.0	20.0	68.0	9.0
MATH	4.3	6.3	32.9	3.0
Four coding tasks	1.0	4.3	18.1	**22.4**
AVERAGE	41.3	42.8	46.5	26.7

Task	Public model	Textbooks Expert	Math Expert	Code Expert	**Weight merging ++**	**Ensem-bling++**
Core 9	68.4	63.0	63.8	38.7	70.6	69.0
MMLU Biology	63.0	66.5	52.5	31.0	60.0	67.5
MMLU CS	52.2	55.0	53.7	36.7	53.8	54.7
MMLU Economics	45.7	49.3	46.0	26.7	47.0	52.7
MMLU Engineering	46.0	51.0	62.0	33.0	56.0	57.0
MMLU Health	59.5	59.8	44.3	30.7	54.4	62.4
MMLU Physics	45.8	49.2	47.7	26.2	47.3	52.5
MMLU Politics	73.0	67.5	62.5	32.0	70.2	71.8
MMLU Pro Biology	56.0	54.0	45.0	34.0	57.0	56.0
MMLU Pro Engineering	18.0	17.0	15.0	10.0	24.0	17.0
AGI Eval	39.0	39.6	40.7	29.0	41.4	43.6
Big Bench Hard	35.6	40.0	45.4	38.2	42.4	43.2
GSM8K	12.0	20.0	68.0	9.0	29.0	12.0
MATH	4.3	6.3	32.9	3.0	6.1	32.9
Four coding tasks	1.0	4.3	18.1	22.4	8.2	23.6
AVERAGE	41.3	42.8	46.5	26.7	44.5	47.7

Task	Public model	Textbooks Expert	Math Expert	Code Expert	Ensem-bling++
Core 9	68.4	63.0	63.8	38.7	69.0
MMLU Biology	63.0	66.5	52.5	31.0	67.5
MMLU CS	52.2	55.0	53.7	36.7	54.7
MMLU Economics	45.7	49.3	46.0	26.7	52.7
MMLU Engineering	46.0	51.0	62.0	33.0	57.0
MMLU Health	59.5	59.8	44.3	30.7	62.4
MMLU Physics	45.8	49.2	47.7	26.2	52.5
MMLU Politics	73.0	67.5	62.5	32.0	71.8
MMLU Pro Biology	56.0	54.0	45.0	34.0	56.0
MMLU Pro Engineering	18.0	17.0	15.0	10.0	17.0
AGI Eval	39.0	39.6	40.7	29.0	43.6
Big Bench Hard	35.6	40.0	45.4	38.2	43.2
GSM8K	12.0	20.0	68.0	9.0	12.0
MATH	4.3	6.3	32.9	3.0	32.9
Four coding tasks	1.0	4.3	18.1	22.4	23.6
AVERAGE	41.3	42.8	46.5	26.7	47.7

Task	Public model	Textbooks Expert	Math Expert	Code Expert	Ensem-bling++	BTX, router trained on shared data
Core 9	68.4	63.0	63.8	38.7	69.0	69.6
MMLU Biology	63.0	66.5	52.5	31.0	67.5	67.0
MMLU CS	52.2	55.0	53.7	36.7	54.7	55.2
MMLU Economics	45.7	49.3	46.0	26.7	52.7	48.0
MMLU Engineering	46.0	51.0	62.0	33.0	57.0	50.0
MMLU Health	59.5	59.8	44.3	30.7	62.4	56.1
MMLU Physics	45.8	49.2	47.7	26.2	52.5	50.2
MMLU Politics	73.0	67.5	62.5	32.0	71.8	70.8
MMLU Pro Biology	56.0	54.0	45.0	34.0	56.0	57.0
MMLU Pro Engineering	18.0	17.0	15.0	10.0	17.0	22.0
AGI Eval	39.0	39.6	40.7	29.0	43.6	43.1
Big Bench Hard	35.6	40.0	45.4	38.2	43.2	41.3
GSM8K	12.0	20.0	68.0	9.0	12.0	27.0
MATH	4.3	6.3	32.9	3.0	32.9	6.7
Four coding tasks	1.0	4.3	18.1	22.4	23.6	6.4
AVERAGE	41.3	42.8	46.5	26.7	47.7	44.7

Task	Public model	Textbooks Expert	Math Expert	Code Expert	Ensem-bling++	BTX, router trained on shared data	**BTX++, router trained on shared data**
Core 9	68.4	63.0	63.8	38.7	69.0	69.6	69.4
MMLU Biology	63.0	66.5	52.5	31.0	67.5	67.0	65.5
MMLU CS	52.2	55.0	53.7	36.7	54.7	55.2	54.8
MMLU Economics	45.7	49.3	46.0	26.7	52.7	48.0	48.7
MMLU Engineering	46.0	51.0	62.0	33.0	57.0	50.0	48.0
MMLU Health	59.5	59.8	44.3	30.7	62.4	56.1	61.1
MMLU Physics	45.8	49.2	47.7	26.2	52.5	50.2	50.0
MMLU Politics	73.0	67.5	62.5	32.0	71.8	70.8	74.0
MMLU Pro Biology	56.0	54.0	45.0	34.0	56.0	57.0	62.0
MMLU Pro Engineering	18.0	17.0	15.0	10.0	17.0	22.0	24.0
AGI Eval	39.0	39.6	40.7	29.0	43.6	43.1	41.4
Big Bench Hard	35.6	40.0	45.4	38.2	43.2	41.3	42.8
GSM8K	12.0	20.0	68.0	9.0	12.0	27.0	38.0
MATH	4.3	6.3	32.9	3.0	32.9	6.7	21.6
Four coding tasks	1.0	4.3	18.1	22.4	23.6	6.4	3.9
AVERAGE	41.3	42.8	46.5	26.7	47.7	44.7	47.0

Task	Public model	Textbooks Expert	Math Expert	Code Expert	Ensem-bling++	BTX, router trained on shared data	BTX++, router trained on shared data	**Ours (no router training)**
Core 9	68.4	63.0	63.8	38.7	69.0	69.6	69.4	70.0
MMLU Biology	63.0	66.5	52.5	31.0	67.5	67.0	65.5	64.0
MMLU CS	52.2	55.0	53.7	36.7	54.7	55.2	54.8	55.5
MMLU Economics	45.7	49.3	46.0	26.7	52.7	48.0	48.7	48.0
MMLU Engineering	46.0	51.0	62.0	33.0	57.0	50.0	48.0	55.0
MMLU Health	59.5	59.8	44.3	30.7	62.4	56.1	61.1	55.6
MMLU Physics	45.8	49.2	47.7	26.2	52.5	50.2	50.0	51.0
MMLU Politics	73.0	67.5	62.5	32.0	71.8	70.8	74.0	69.5
MMLU Pro Biology	56.0	54.0	45.0	34.0	56.0	57.0	62.0	54.0
MMLU Pro Engineering	18.0	17.0	15.0	10.0	17.0	22.0	24.0	24.0
AGI Eval	39.0	39.6	40.7	29.0	43.6	43.1	41.4	41.1
Big Bench Hard	35.6	40.0	45.4	38.2	43.2	41.3	42.8	44.9
GSM8K	12.0	20.0	68.0	9.0	12.0	27.0	38.0	65.0
MATH	4.3	6.3	32.9	3.0	32.9	6.7	21.6	24.9
Four coding tasks	1.0	4.3	18.1	22.4	23.6	6.4	3.9	16.6
AVERAGE	41.3	42.8	46.5	26.7	47.7	44.7	47.0	49.3

Task	Public model	Textbooks Expert	Math Expert	Code Expert	Ensem-bling++	BTX, router trained on shared data	BTX++, router trained on shared data	Ours (no router training)	**Ours (router trained on proxy data)**
Core 9	68.4	63.0	63.8	38.7	69.0	69.6	69.4	70.0	**72.1**
MMLU Biology	63.0	66.5	52.5	31.0	67.5	67.0	65.5	64.0	**67.5**
MMLU CS	52.2	55.0	53.7	36.7	54.7	55.2	54.8	55.5	**60.0**
MMLU Economics	45.7	49.3	46.0	26.7	52.7	48.0	48.7	48.0	**55.7**
MMLU Engineering	46.0	51.0	62.0	33.0	57.0	50.0	48.0	55.0	**62.0**
MMLU Health	59.5	59.8	44.3	30.7	62.4	56.1	61.1	55.6	**62.6**
MMLU Physics	45.8	49.2	47.7	26.2	52.5	50.2	50.0	51.0	**53.2**
MMLU Politics	73.0	67.5	62.5	32.0	71.8	70.8	74.0	69.5	**74.2**
MMLU Pro Biology	56.0	54.0	45.0	34.0	56.0	57.0	62.0	54.0	**63.0**
MMLU Pro Engineering	18.0	17.0	15.0	10.0	17.0	22.0	24.0	24.0	**26.0**
AGI Eval	39.0	39.6	40.7	29.0	43.6	43.1	41.4	41.1	**46.1**
Big Bench Hard	35.6	40.0	45.4	38.2	43.2	41.3	42.8	44.9	**47.4**
GSM8K	12.0	20.0	**68.0**	9.0	12.0	27.0	38.0	65.0	63.0
MATH	4.3	6.3	**32.9**	3.0	32.9	6.7	21.6	24.9	24.1
Four coding tasks	1.0	4.3	18.1	**22.4**	23.6	6.4	3.9	16.6	10.3
AVERAGE	41.3	42.8	46.5	26.7	47.7	44.7	47.0	49.3	**52.5**

Task	Public model	Textbooks Expert	Math Expert	Code Expert	Ensem-bling++	BTX, router trained on shared data	BTX++, router trained on shared data	Ours (no router training)	**Ours (router trained on proxy data)**
Core 9	68.4	63.0	63.8	38.7	69.0	69.6	69.4	70.0	**72.1**
MMLU Biology	63.0	66.5	52.5	31.0	67.5	67.0	65.5	64.0	**67.5**
MMLU CS	52.2	55.0	53.7	36.7	54.7	55.2	54.8	55.5	**60.0**
MMLU Economics	45.7	49.3	46.0	26.7	52.7	48.0	48.7	48.0	**55.7**
MMLU Engineering	46.0	51.0	62.0	33.0	57.0	50.0	48.0	55.0	**62.0**
MMLU Health	59.5	59.8	44.3	30.7	62.4	56.1	61.1	55.6	**62.6**
MMLU Physics	45.8	49.2	47.7	26.2	52.5	50.2	50.0	51.0	**53.2**
MMLU Politics	73.0	67.5	62.5	32.0	71.8	70.8	74.0	69.5	**74.2**
MMLU Pro Biology	56.0	54.0	45.0	34.0	56.0	57.0	62.0	54.0	**63.0**
MMLU Pro Engineering	18.0	17.0	15.0	10.0	17.0	22.0	24.0	24.0	**26.0**
AGI Eval	39.0	39.6	40.7	29.0	43.6	43.1	41.4	41.1	**46.1**
Big Bench Hard	35.6	40.0	45.4	38.2	43.2	41.3	42.8	44.9	**47.4**
GSM8K	12.0	20.0	**68.0**	9.0	12.0	27.0	38.0	65.0	63.0
MATH	4.3	6.3	**32.9**	3.0	32.9	6.7	21.6	24.9	24.1
Four coding tasks	1.0	4.3	18.1	**22.4**	23.6	6.4	3.9	16.6	10.3
AVERAGE	41.3	42.8	46.5	26.7	47.7	44.7	47.0	49.3	**52.5**
								(+20%)	(+27%)

Largest gains on benchmarks benefiting from new data sources 😃

Task	Public model	Textbooks Expert	Math Expert	Code Expert	Ensem-bling++	BTX, router trained on shared data	BTX++, router trained on shared data	Ours (no router training)	**Ours (router trained on proxy data)**
Core 9	68.4	63.0	63.8	38.7	69.0	69.6	69.4	70.0	**72.1**
MMLU Biology	63.0	66.5	52.5	31.0	67.5	67.0	65.5	64.0	**67.5**
MMLU CS	52.2	55.0	53.7	36.7	54.7	55.2	54.8	55.5	**60.0**
MMLU Economics	45.7	49.3	46.0	26.7	52.7	48.0	48.7	48.0	**55.7**
MMLU Engineering	46.0	51.0	62.0	33.0	57.0	50.0	48.0	55.0	**62.0**
MMLU Health	59.5	59.8	44.3	30.7	62.4	56.1	61.1	55.6	**62.6**
MMLU Physics	45.8	49.2	47.7	26.2	52.5	50.2	50.0	51.0	**53.2**
MMLU Politics	73.0	67.5	62.5	32.0	71.8	70.8	74.0	69.5	**74.2**
MMLU Pro Biology	56.0	54.0	45.0	34.0	56.0	57.0	62.0	54.0	**63.0**
MMLU Pro Engineering	18.0	17.0	15.0	10.0	17.0	22.0	24.0	24.0	**26.0**
AGI Eval	39.0	39.6	40.7	29.0	43.6	43.1	41.4	41.1	**46.1**
Big Bench Hard	35.6	40.0	45.4	38.2	43.2	41.3	42.8	44.9	**47.4**
GSM8K	12.0	20.0	**68.0**	9.0	12.0	27.0	38.0	65.0	63.0
MATH	4.3	6.3	**32.9**	3.0	32.9	6.7	21.6	24.9	24.1
Four coding tasks	1.0	4.3	18.1	**22.4**	23.6	6.4	3.9	16.6	10.3
AVERAGE	41.3	42.8	46.5	26.7	47.7	44.7	47.0	49.3 (+20%)	**52.5** (+27%)

Significant gains even on benchmarks with no single data source helped 😃

Task	Public model	Textbooks Expert	Math Expert	Code Expert	Ensem-bling++	BTX, router trained on shared data	BTX++, router trained on shared data	Ours (no router training)	**Ours (router trained on proxy data)**
Core 9	68.4	63.0	63.8	38.7	69.0	69.6	69.4	70.0	**72.1**
MMLU Biology	63.0	66.5	52.5	31.0	67.5	67.0	65.5	64.0	**67.5**
MMLU CS	52.2	55.0	53.7	36.7	54.7	55.2	54.8	55.5	**60.0**
MMLU Economics	45.7	49.3	46.0	26.7	52.7	48.0	48.7	48.0	**55.7**
MMLU Engineering	46.0	51.0	62.0	33.0	57.0	50.0	48.0	55.0	**62.0**
MMLU Health	59.5	59.8	44.3	30.7	62.4	56.1	61.1	55.6	**62.6**
MMLU Physics	45.8	49.2	47.7	26.2	52.5	50.2	50.0	51.0	**53.2**
MMLU Politics	73.0	67.5	62.5	32.0	71.8	70.8	74.0	69.5	**74.2**
MMLU Pro Biology	56.0	54.0	45.0	34.0	56.0	57.0	62.0	54.0	**63.0**
MMLU Pro Engineering	18.0	17.0	15.0	10.0	17.0	22.0	24.0	24.0	**26.0**
AGI Eval	39.0	39.6	40.7	29.0	43.6	43.1	41.4	41.1	**46.1**
Big Bench Hard	35.6	40.0	45.4	38.2	43.2	41.3	42.8	44.9	**47.4**
GSM8K	12.0	20.0	**68.0**	9.0	12.0	27.0	38.0	65.0	63.0
MATH	4.3	6.3	**32.9**	3.0	32.9	6.7	21.6	24.9	24.1
Four coding tasks	1.0	4.3	18.1	**22.4**	23.6	6.4	3.9	16.6	10.3
AVERAGE	41.3	42.8	46.5	26.7	47.7	44.7	47.0	49.3	**52.5**
								(+20%)	(+27%)

Specialized tasks: Better than prev public model, lags behind specialized experts 🤔

Task	Public model	Textbooks Expert	Math Expert	Code Expert	Ensem-bling++	BTX, router trained on shared data	BTX++, router trained on shared data	Ours (no router training)	**Ours (router trained on proxy data)**
Core 9	68.4	63.0	63.8	38.7	69.0	69.6	69.4	70.0	**72.1**
MMLU Biology	63.0	66.5	52.5	31.0	67.5	67.0	65.5	64.0	**67.5**
MMLU CS	52.2	55.0	53.7	36.7	54.7	55.2	54.8	55.5	**60.0**
MMLU Economics	45.7	49.3	46.0	26.7	52.7	48.0	48.7	48.0	**55.7**
MMLU Engineering	46.0	51.0	62.0	33.0	57.0	50.0	48.0	55.0	**62.0**
MMLU Health	59.5	59.8	44.3	30.7	62.4	56.1	61.1	55.6	**62.6**
MMLU Physics	45.8	49.2	47.7	26.2	52.5	50.2	50.0	51.0	**53.2**
MMLU Politics	73.0	67.5	62.5	32.0	71.8	70.8	74.0	69.5	**74.2**
MMLU Pro Biology	56.0	54.0	45.0	34.0	56.0	57.0	62.0	54.0	**63.0**
MMLU Pro Engineering	18.0	17.0	15.0	10.0	17.0	22.0	24.0	24.0	**26.0**
AGI Eval	39.0	39.6	40.7	29.0	43.6	43.1	41.4	41.1	**46.1**
Big Bench Hard	35.6	40.0	45.4	38.2	43.2	41.3	42.8	44.9	**47.4**
GSM8K	12.0	20.0	**68.0**	9.0	12.0	27.0	38.0	65.0	63.0
MATH	4.3	6.3	**32.9**	3.0	32.9	6.7	21.6	24.9	24.1
Four coding tasks	1.0	4.3	18.1	**22.4**	23.6	6.4	3.9	16.6	10.3
AVERAGE	41.3	42.8	46.5	26.7	47.7	44.7	47.0	49.3 (+20%)	**52.5** (+27%)

Summary

Summary

Goal

To make use of data with restrictions

- Siloed datasets
- Train in isolation
- Resulting model should be a public, general-purpose model

Summary

Goal

To make use of data with restrictions

- Siloed datasets
- Train in isolation
- Resulting model should be a public, general-purpose model

Method

Started with (variants of) existing approaches

+ Two key ideas

- MoE-aware siloed training
- Obtaining proxy data for router training

Summary

Goal

To make use of data with restrictions

- Siloed datasets
- Train in isolation
- Resulting model should be a public, general-purpose model

Method

Started with (variants of) existing approaches

+ Two key ideas

- MoE-aware siloed training
- Obtaining proxy data for router training

Results

On real proprietary or simulated proprietary data in realistic domains

- 27% relative gain over the prev public model
- 20% relative gain even without router training

Open problems

1. Can a combined model beat specialized experts in specialized tasks?
2. Fine-grained data & scaling # of number of datasets
3. Can we completely remove router training? (Nonparametric router)

Open problems

1. Can a combined model beat specialized experts in specialized tasks?
2. Fine-grained data & scaling # of number of datasets
3. Can we completely remove router training? (Nonparametric router)

This may end up looking like a retrieval model 🤔

Thank you for listening!

sewonmin.com

sewonm@berkeley.edu

Please leave feedback at tinyurl.com/sewonm-talk