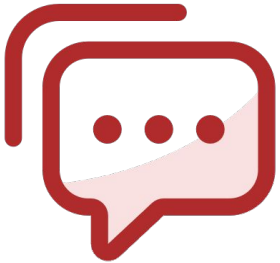
slido

Please download and install the Slido app on all computers you use



Audience Q&A

ⓘ Start presenting to display the audience questions on this slide.

Announcements

HW 5 is out.

- It is designed to be a light weight one for you to get ready for HW6

HW6 is a mini-project of two people team

- Pick something you learned in this class and applied it in your scenario
- To submit: A project report and source code
- Project presentation in the last two lectures
- Start forming team now!

Carnegie Mellon University

Efficient Pretraining with Sparse Models

Large Language Models: Methods and Applications

Daphne Ippolito and Chenyan Xiong

Learning Objectives

Learn the basics of Mixture-of-Expert Models

Understand various challenges of Mixture-of-Experts and methods to address them

Learn initial knowledge on when to use Mixture-of-Experts

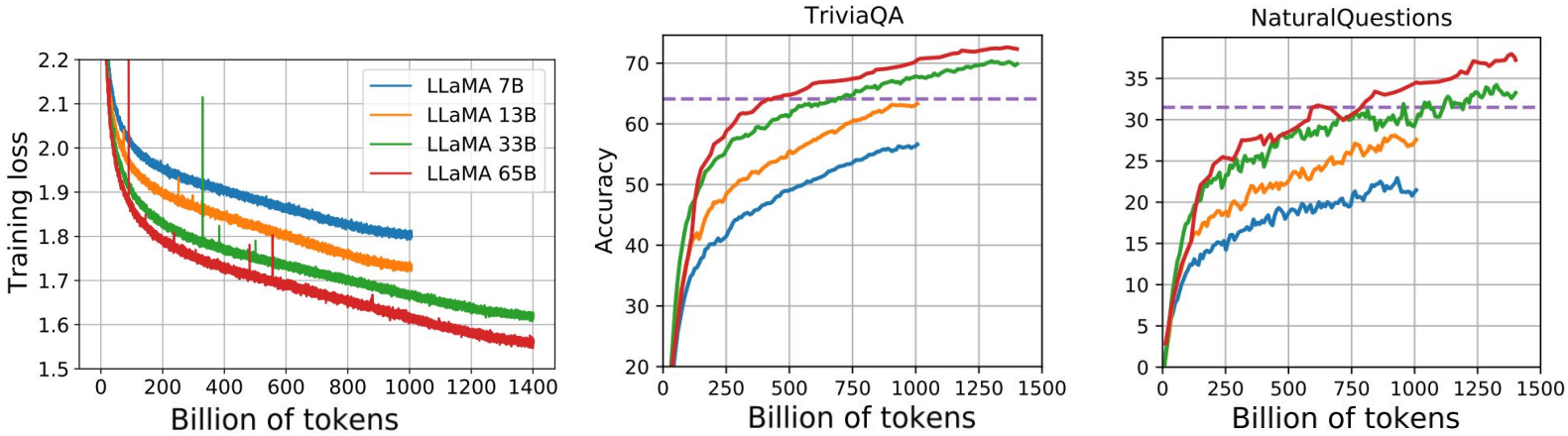
Outline

Mixture-of-Experts (MoE)

- Motivation
- Standard Configuration
- Expert Routing
- Expert Specialization
- Remarks

Why Sparsity?

The power of deep learning comes from over-parameterization.

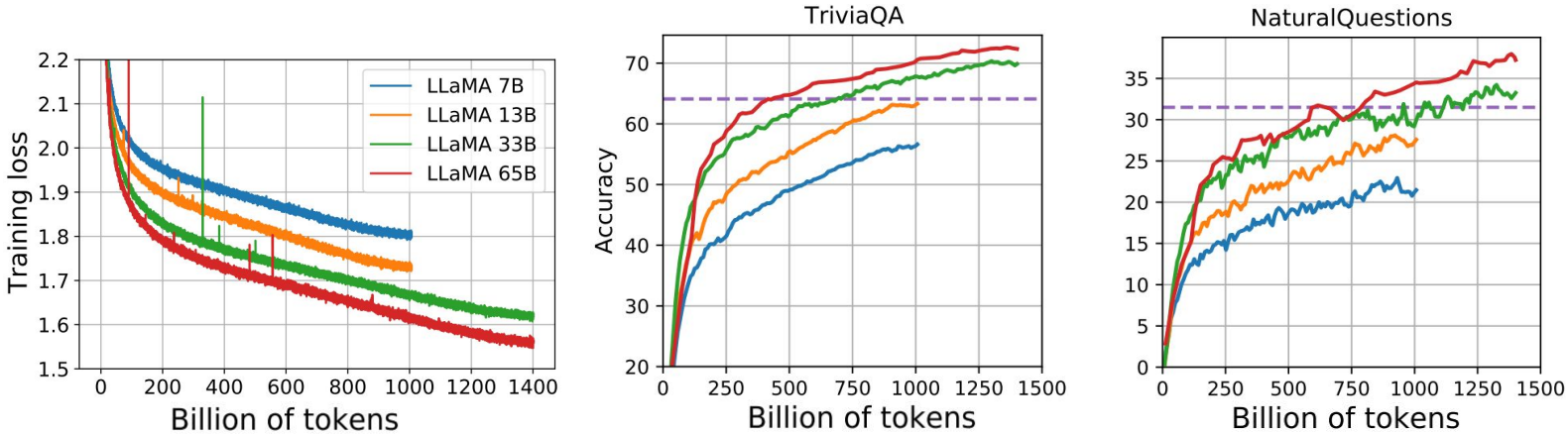


Scaling up LLMs Exponentially Leads to Linear Accuracy Improvements [1]

[1] Touvron et al. 2023. LLaMA: Open and Efficient Foundation Language Models.

Why Sparsity?

The power of deep learning comes from over-parameterization.



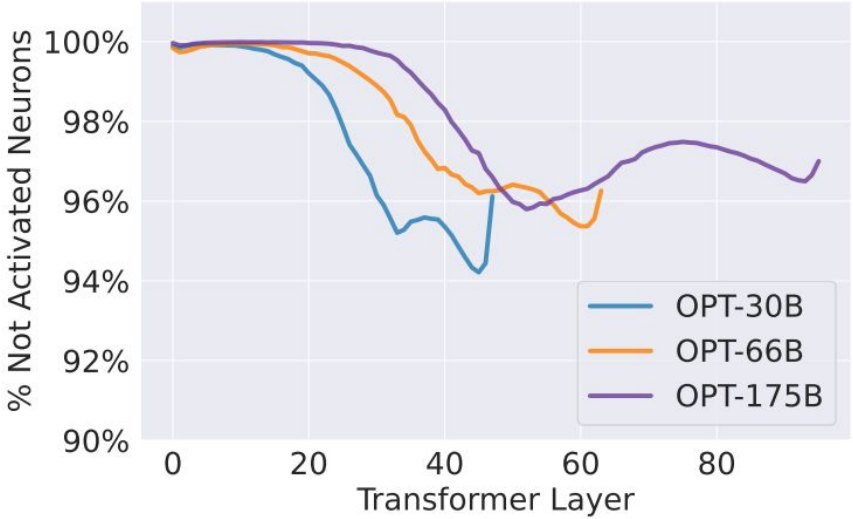
Scaling up LLMs Exponentially Leads to Linear Accuracy Improvements [1]

Is it the best utilization of parameters?

[1] Touvron et al. 2023. LLaMA: Open and Efficient Foundation Language Models.

Why Sparsity?

Large language models are sparse, everywhere.



Sparsity in Pretrained OPT models [2]

[2] Liu et al. 2023. Deja Vu: Contextual Sparsity for Efficient LLMs at Inference Time

Why Sparsity?

Sparsity of LLMs has been widely adopted in many applications:

- Pruning
- Distillation
- Efficient Serving
- Long-context Models

Why Sparsity?

Sparsity of LLMs has been widely adopted in many applications:

- Pruning
- Distillation
- Efficient Serving
- Long-context Models

All as an “after thought”:

- First pretrain a huge dense, but sparse, model
- Then make it more efficient through sparsity

Why Sparsity?

Sparsity of LLMs has been widely adopted in many applications:

- Pruning
- Distillation
- Efficient Serving
- Long-context Models

All as an "after thought":

- First pretrain a huge dense, but sparse, model
- Then make it more efficient through sparsity

Can we leverage sparsity for efficient large scale pretraining?

Mixture-of-Experts to Leverage Sparisity

Conditional Computation for sparsity

- Only activate a part of the neural network
- But a different part of it conditioned on inputs

Mixture-of-Experts to Leverage Sparisity

Conditional Computation for sparsity

- Only activate a part of the neural network
- But a different part of it conditioned on inputs

Conditioned on tokens

- Each token activates different parts of the network

Mixture-of-Experts to Leverage Sparisity

Conditional Computation for sparsity

- Only activate a part of the neural network
- But a different part of it conditioned on inputs

Conditioned on tokens

- Each token activates different parts of the network

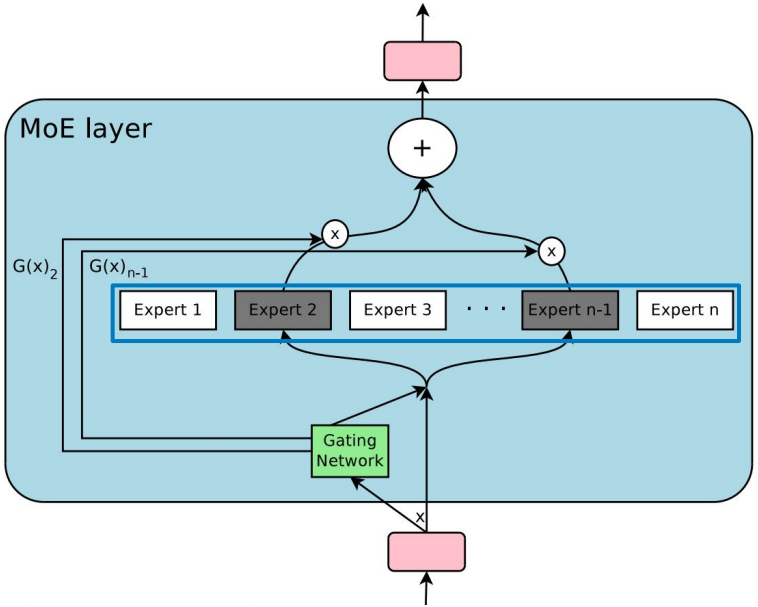
Applied on the FFN layers

- Focus on making the FFNs sparse, the simplest and the heaviest part of LLMs

Sparsely-Gated Mixture-of-Experts Layer

- Split one FFN into multiple (N) smaller FFNs (and name them experts)

$$E_i(x) = \mathrm{FFN}(x)$$



Sparsely-Gated MoE Layer [3]

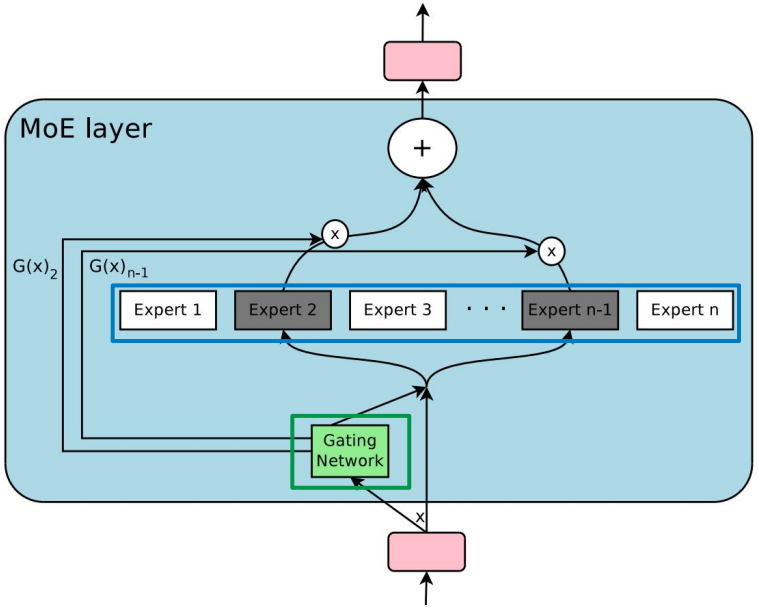
Sparsely-Gated Mixture-of-Experts Layer

- Split one FFN into multiple (N) smaller FFNs (and name them experts)

$$E_i(x) = \text{FFN}(x)$$

Train a gating function to assign weights on experts

$$G'(x) = \text{softmax}\left(x \cdot W_g\right)$$



Sparsely-Gated MoE Layer [3]

Sparsely-Gated Mixture-of-Experts Layer

- Split one FFN into multiple (N) smaller FFNs (and name them experts)

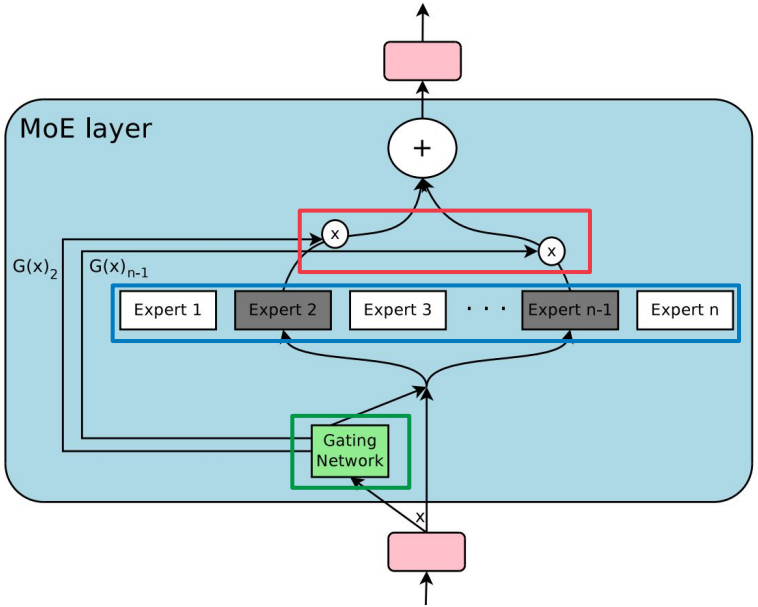
$$E_i(x) = \mathrm{FFN}(x)$$

Train a gating function to assign weights on experts

$$G'(x) = \mathrm{softmax}\left(x \cdot W_g\right)$$

Pick top k experts to activate for this input

$$G(x) = \mathrm{TopK}(G'(x))$$



Sparsely-Gated MoE Layer [3]

Sparsely-Gated Mixture-of-Experts Layer

- Split one FFN into multiple (N) smaller FFNs (and name them experts)

$$E_i(x) = \mathrm{FFN}(x)$$

Train a gating function to assign weights on experts

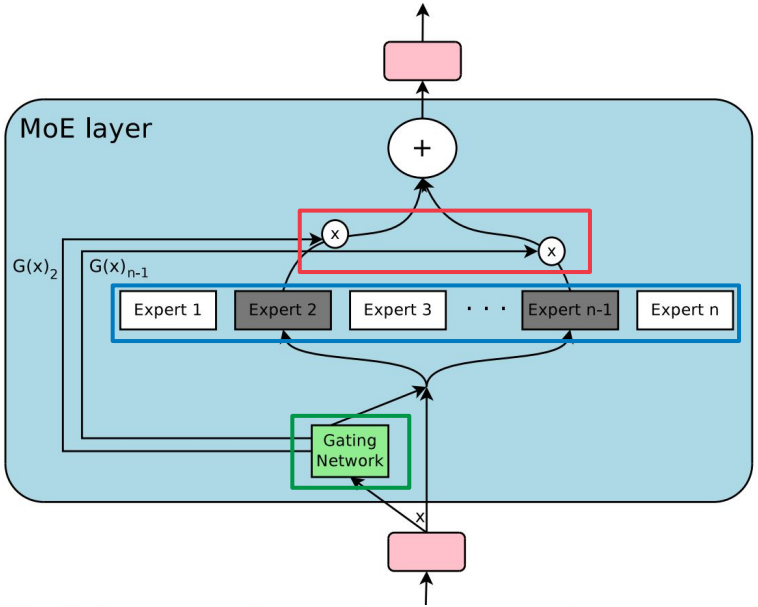
$$G'(x) = \mathrm{softmax}\left(x \cdot W_g\right)$$

Pick top k experts to activate for this input

$$G(x) = \mathrm{TopK}(G'(x))$$

Resulted in activated small FFNs per token:

K/N e.g., 2/64

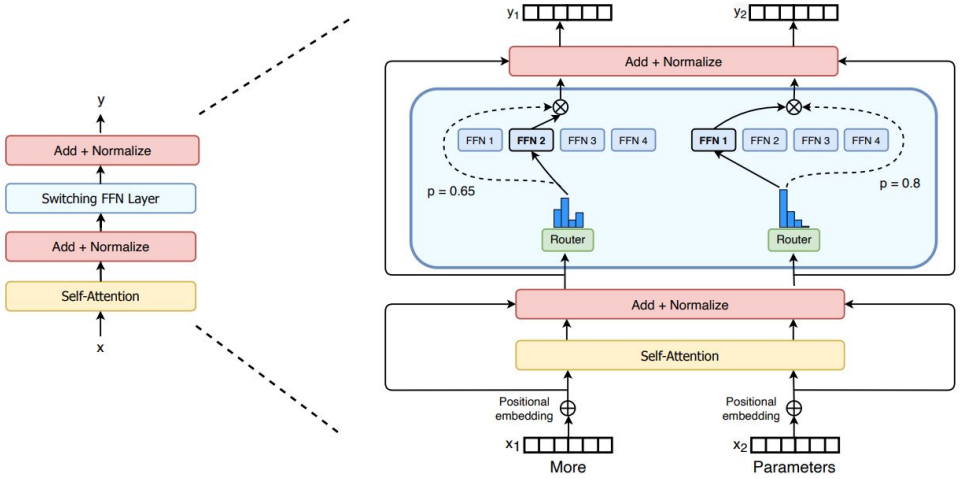


Sparsely-Gated MoE Layer [3]

Switch Transformers

Switch FFN layers in Transformers to MoE layers

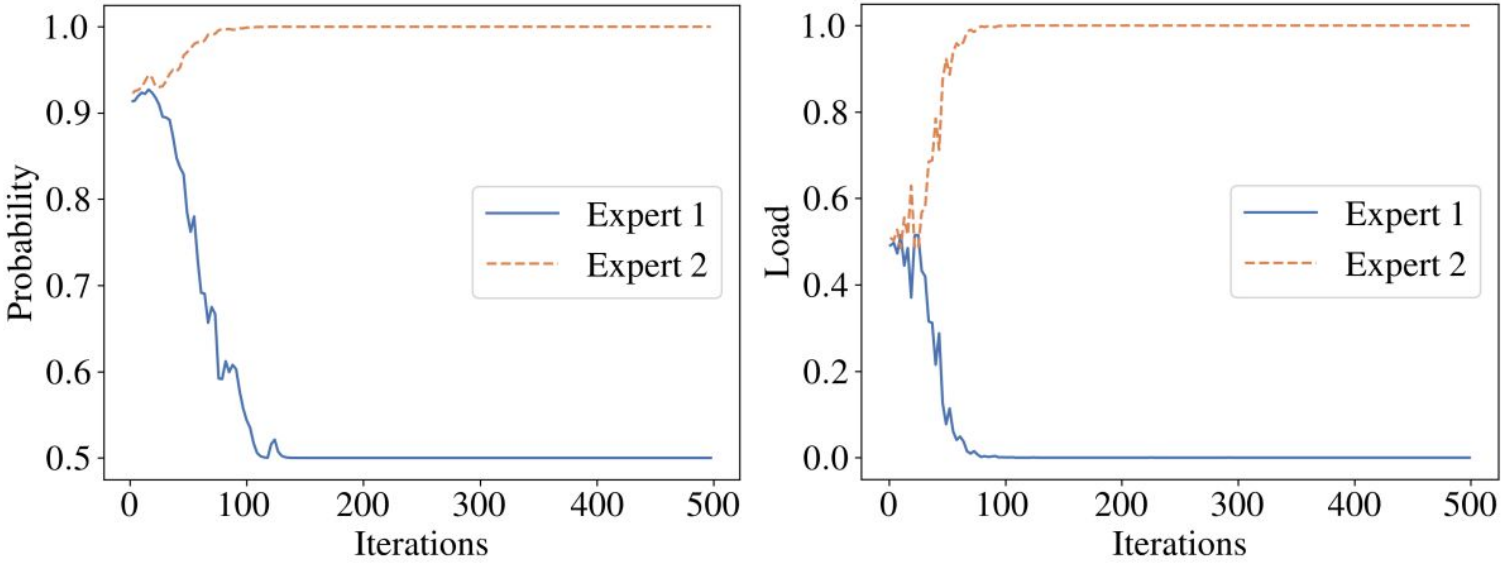
- Mixing the switch layers with dense FFNs in the Transformer
- E.g., every other layer



Switch Transformer [4]

Expert Degeneration

What if all tokens are routed to one expert?



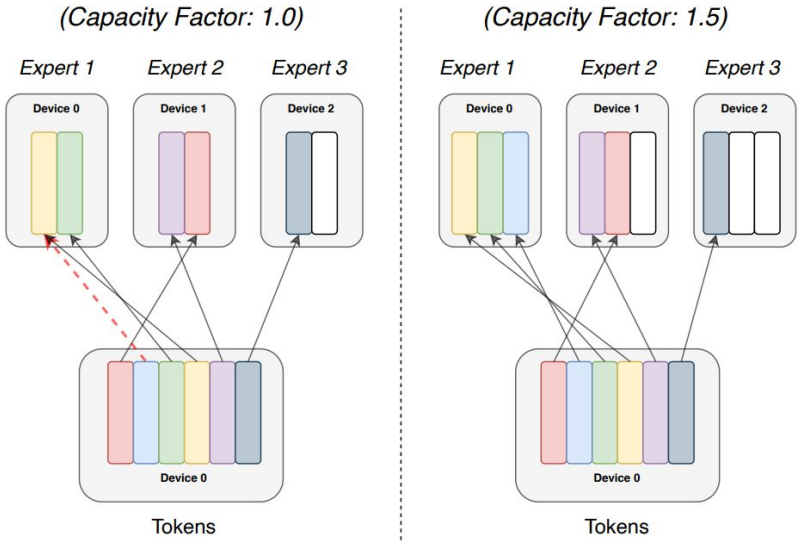
Expert Degeneration in Two Expert Language Models [5]

[5] Zuo et al. 2022. Taming Sparsely Activated Transformer with Stochastic Experts

Switch Transformers: Expert Load Balacing

- Restrict the maximum tokens an expert can take in a batch

$$\text{Expert Capacity} = \left(\frac{\text{Tokens per batch}}{\text{number of experts}}\right) \times \text{Capacity Factor}$$



Expert Capacity Capping [4]

Switch Transformers: Expert Load Balacing

- Restrict the maximum tokens an expert can take in a batch

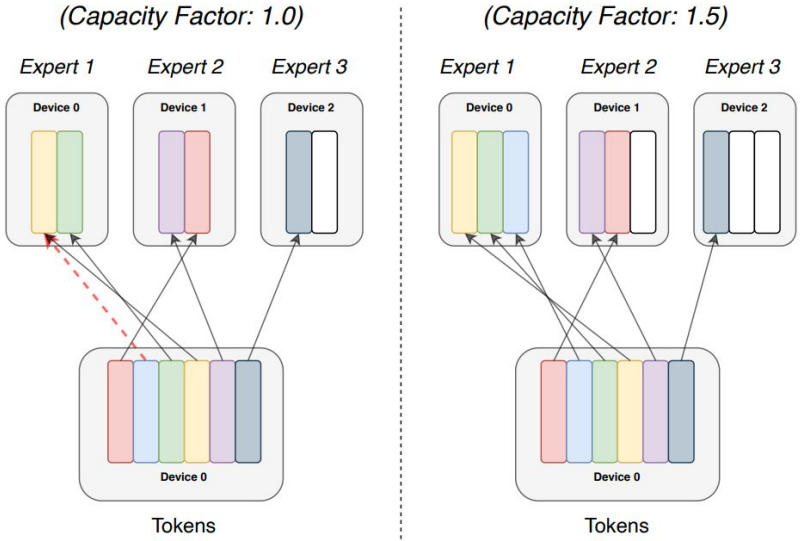
$$\text{Expert Capacity} = \left(\frac{\text{Tokens per batch}}{\text{number of experts}}\right) \times \text{Capacity Factor}$$

Capacity Factor = 1:

- Absolute Uniform Allocation

Capacity Factor = 1.5:

- An expert can take 1.5x token #



Expert Capacity Capping [4]

Switch Transformers: Expert Load Balacing

- Restrict the maximum tokens an expert can take in a batch

$$\text{Expert Capacity} = \left(\frac{\text{Tokens per batch}}{\text{number of experts}}\right) \times \text{Capacity Factor}$$

Capacity Factor = 1:

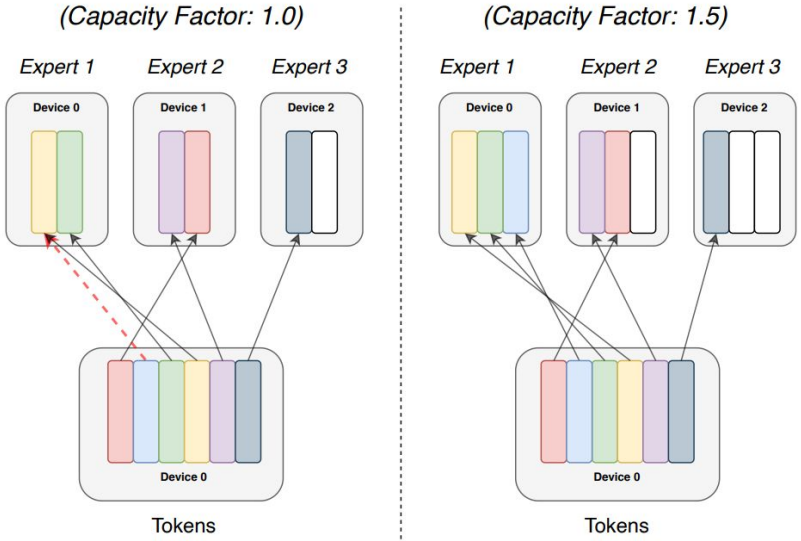
- Absolute Uniform Allocation

Capacity Factor = 1.5:

- An expert can take 1.5x token #

What if token load > Expert Capacity?

- Skip this layer (lol)



Expert Capacity Capping [4]

Switch Transformers: Expert Load Balacing

- Restrict the maximum tokens an expert can take in a batch

$$\text{Expert Capacity} = \left(\frac{\text{Tokens per batch}}{\text{number of experts}}\right) \times \text{Capacity Factor}$$

Capacity Factor = 1:

- Absolute Uniform Allocation

Capacity Factor = 1.5:

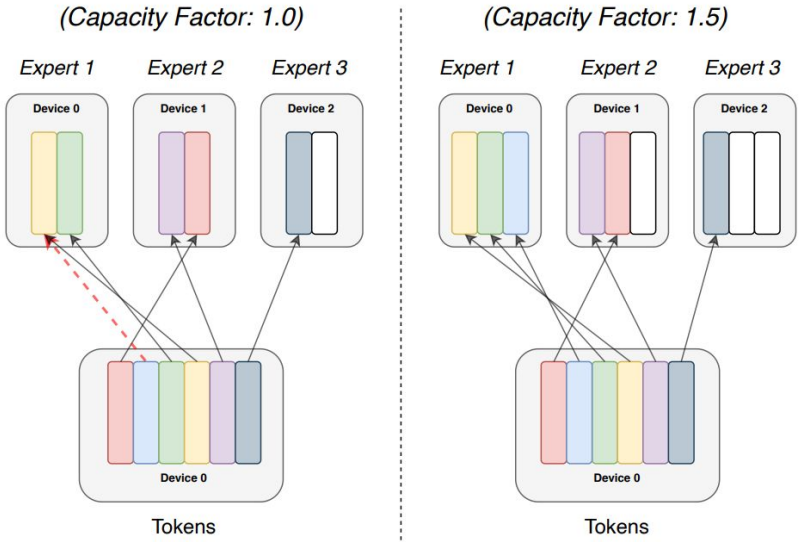
- An expert can take 1.5x token #

What if token load > Expert Capacity?

- Skip this layer (lol)

Plus, additional load balancing loss:

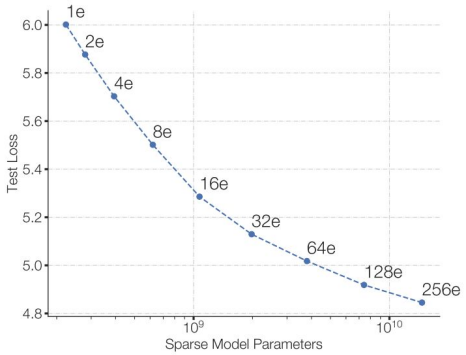
$$l = N \sum_{i=1}^{N} \frac{\text{Tokens assigned to i}}{\text{Total Token Count}} \times \sum_x G'(x)$$



Expert Capacity Capping [4]

Switch Transformers: Empirical Performance

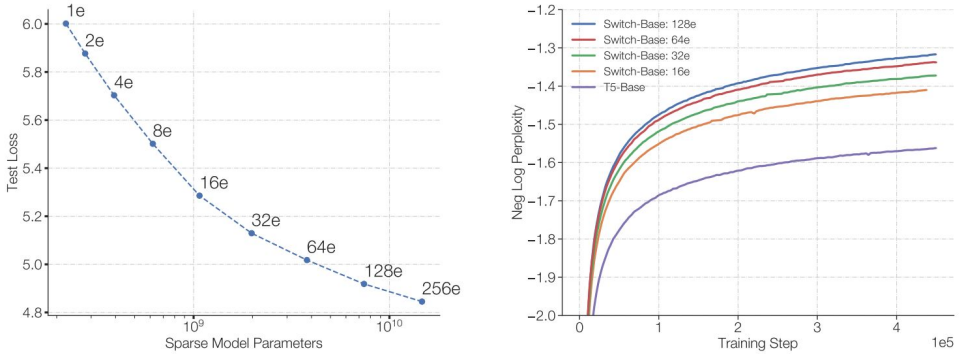
Performance by splitting FFN in T5 to different number of experts (1e-256e)



Language Modeling Loss on C4 [4]

Switch Transformers: Empirical Performance

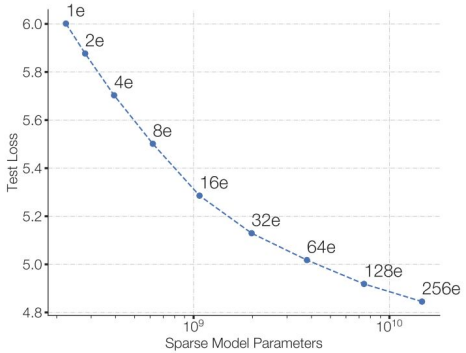
Performance by splitting FFN in T5 to different number of experts (1e-256e)

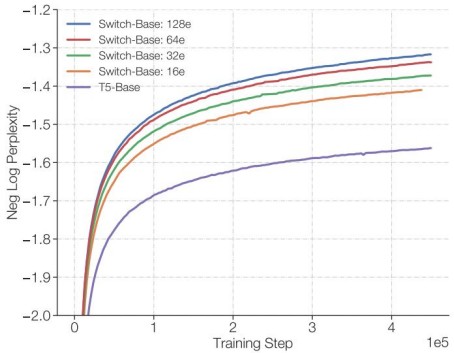


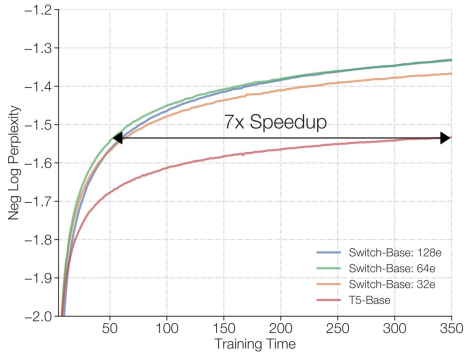
Language Modeling Loss on C4 [4]

Switch Transformers: Empirical Performance

Performance by splitting FFN in T5 to different number of experts (1e-256e)



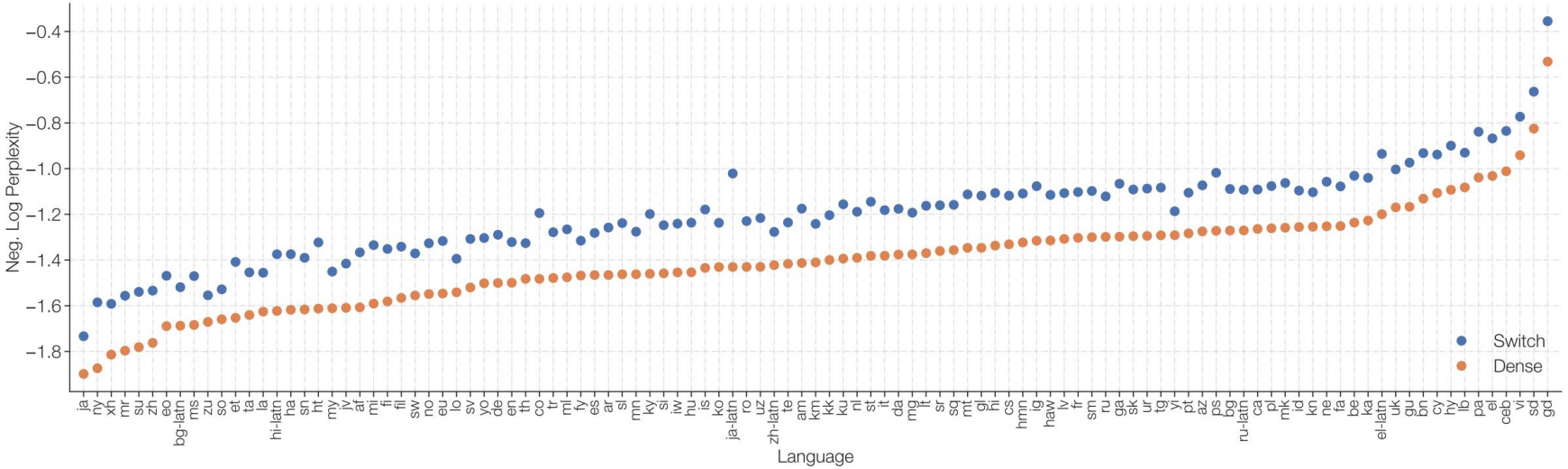




Language Modeling Loss on C4 [4]

Switch Transformers: Empirical Performance

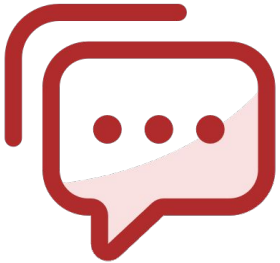
Performance by splitting FFN in T5 to different number of experts (1e-256e)



Machine Translation Performance [4]

slido

Please download and install the Slido app on all computers you use



Audience Q&A

ⓘ Start presenting to display the audience questions on this slide.

Outline

Mixture-of-Experts (MoE)

- Motivation
- Standard Configuration
- **Expert Routing**
- Expert Specialization
- Remarks

MoE: Challenges

Expert Routing

- How to match tokens and experts for both balanced load and maximum effectiveness?

Expert Specialization

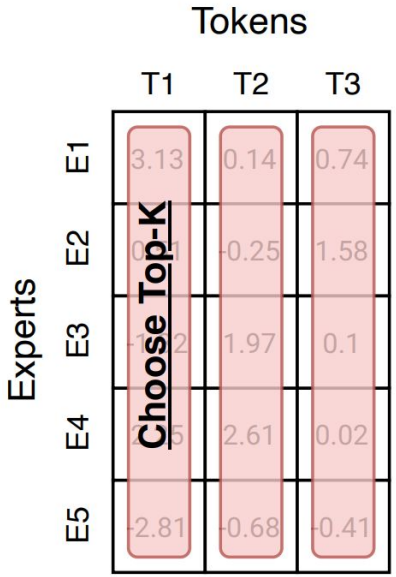
- Are experts really “experts” on some specialized capability? Or it is just ensemble?

Infrastructure Efficiency

- Unique challenges in training and serving

MoE: Expert Routing

How to match tokens and experts?

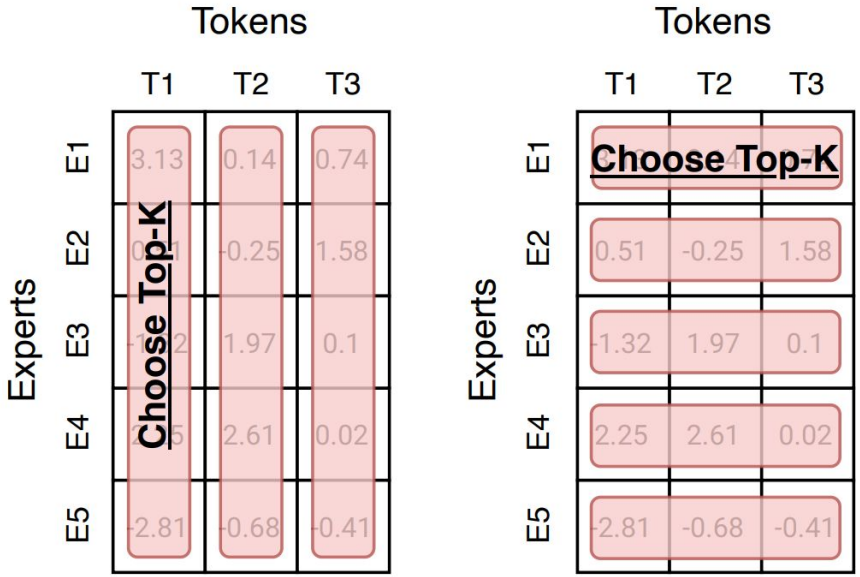


Three Ways to Match Token-Expert [5]

[5] Fedus et al. 2022. A Review of Sparse Expert Models in Deep Learning

MoE: Expert Routing

How to match tokens and experts?

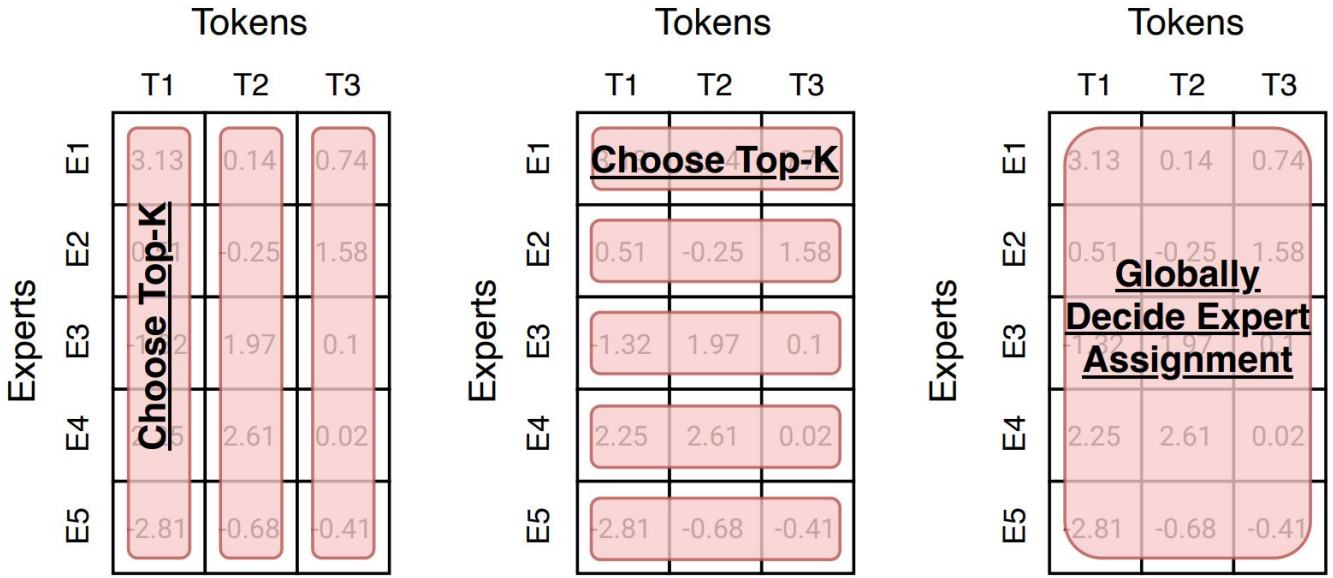


Three Ways to Match Token-Expert [5]

[5] Fedus et al. 2022. A Review of Sparse Expert Models in Deep Learning

MoE: Expert Routing

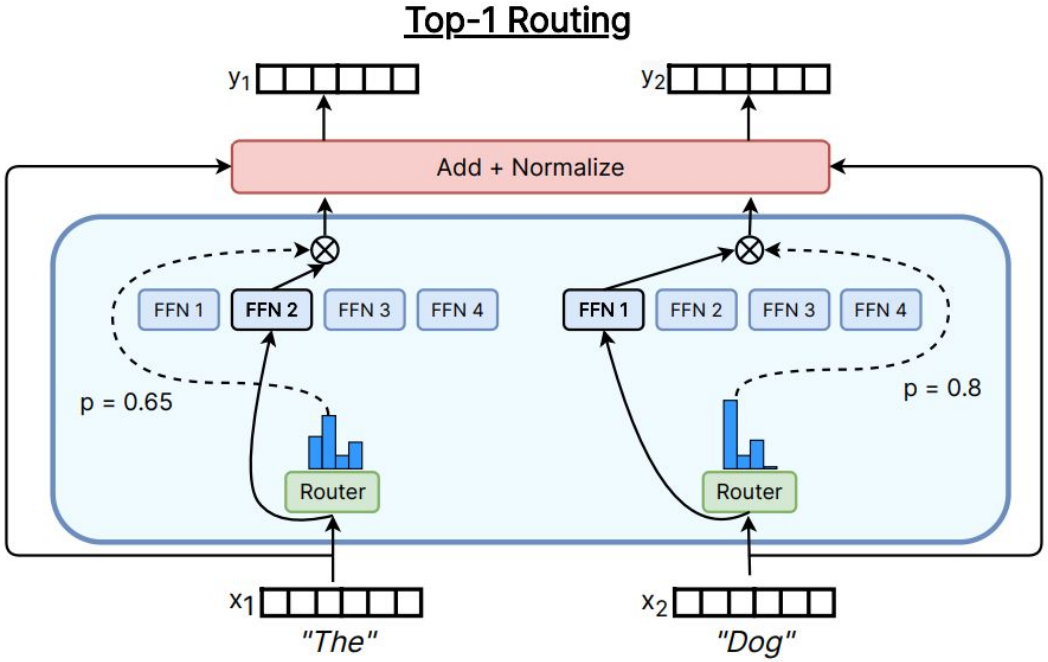
How to match tokens and experts?



Three Ways to Match Token-Expert [5]

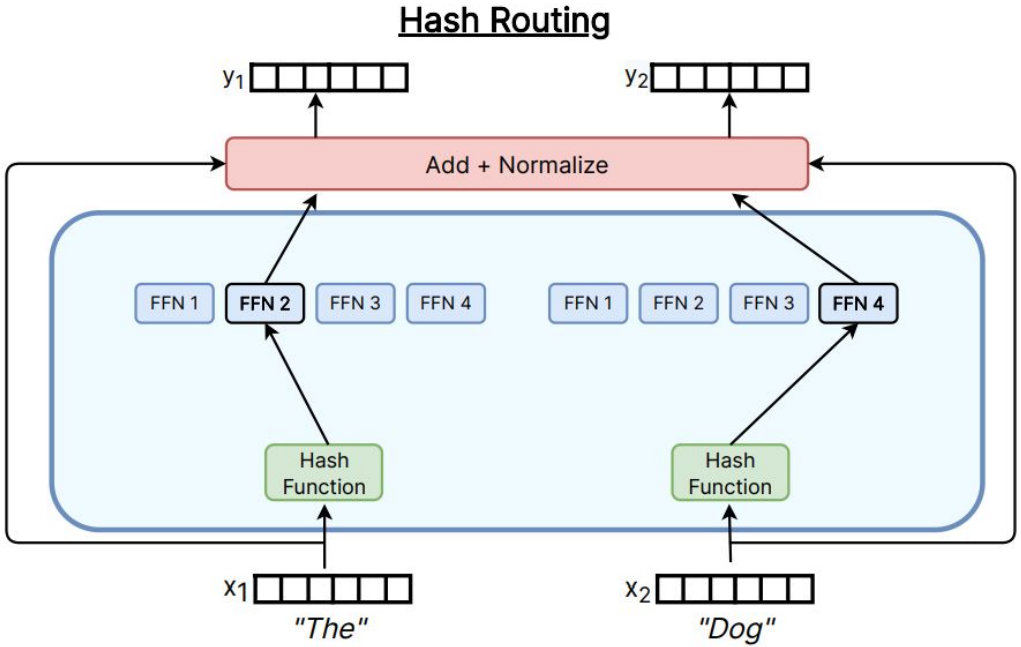
MoE: Expert Routing

Each token goes to its top experts learned from the gating layer



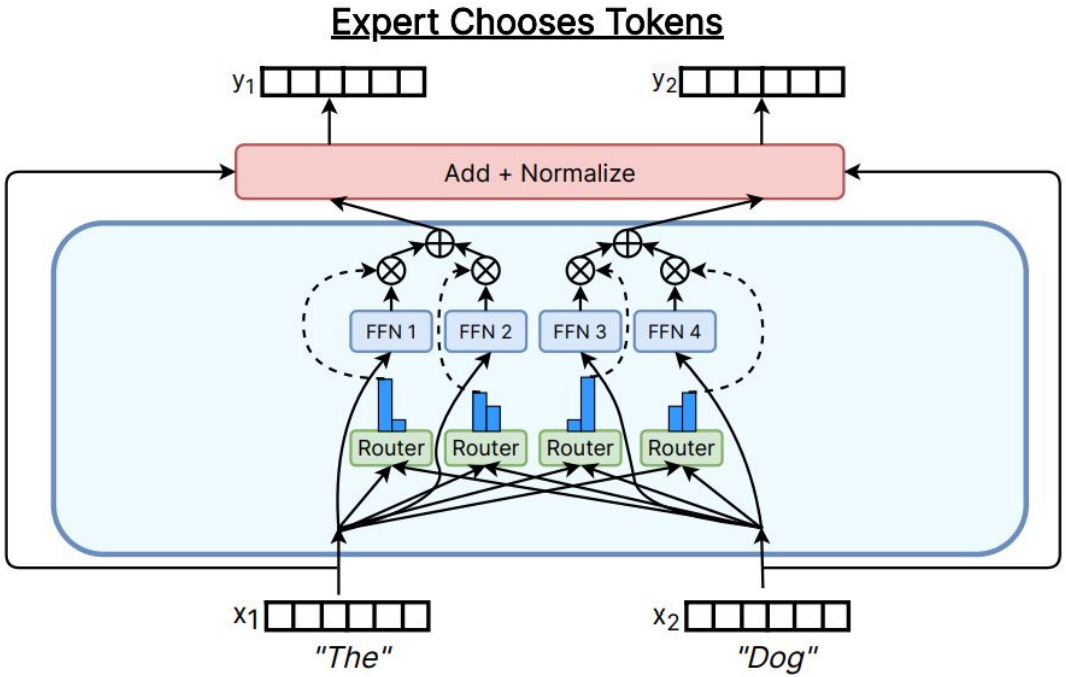
MoE: Expert Routing

Each token goes to a deterministic expert using a hash function



MoE: Expert Routing

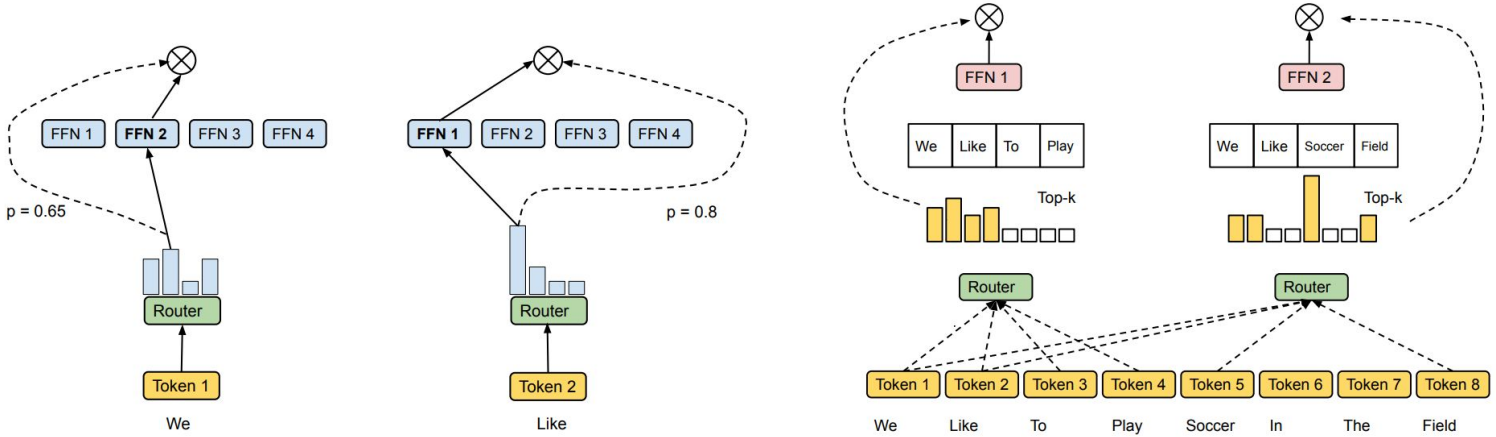
Expert chooses its top K tokens



Expert Choice Routing

Using a gating layer per expert to learn the top tokens it selects

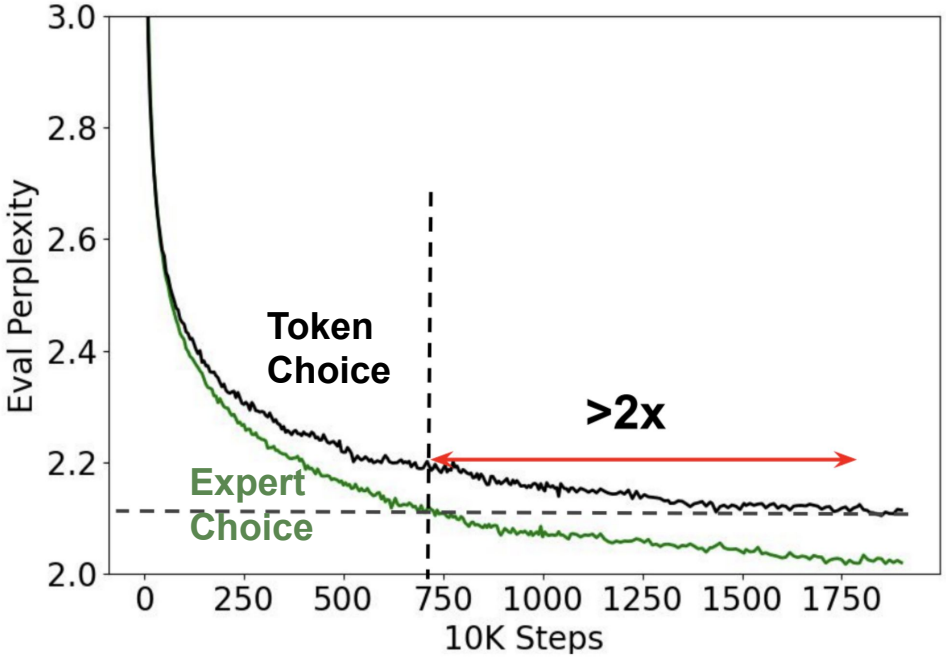
- [optional] add a constraints on maximum number of experts allowed per token



Comparison of token choice and expert choice routing [6]

Expert Choice Routing: Performance

Faster convergence versus token choice routing



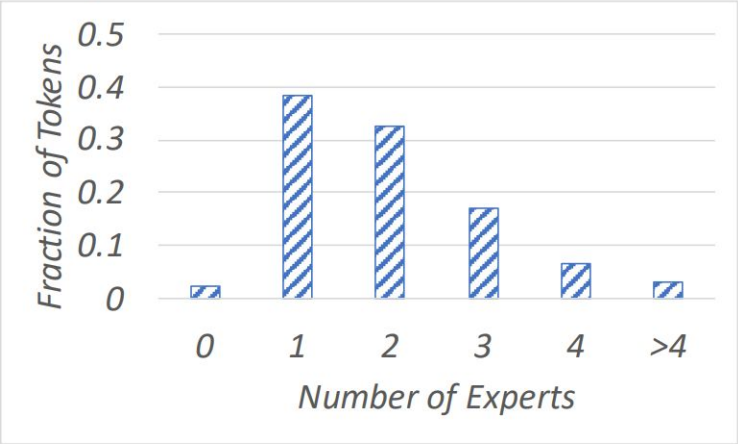
Language Model Loss with Different Routing [6]

[6] Zhou et al. 2022. Mixture-of-Experts with Expert Choice Routing

Expert Choice Routing: Performance

Learns to assign variant number of experts per token

- Important/informative tokens allowed more capacity
- Non informative tokens with reduced capacity

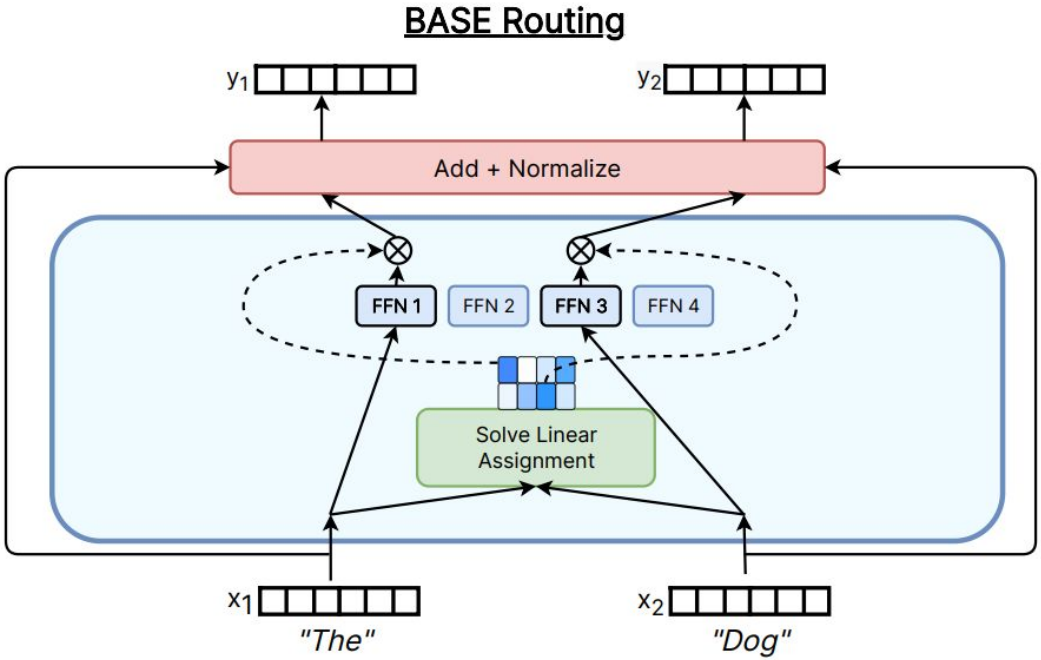


Number of Experts Assigned Per Token [6]

MoE: Expert Routing

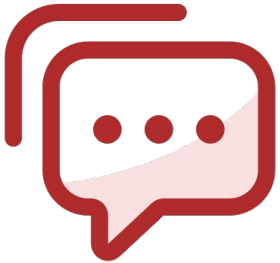
Can be fancier with global matching of token-experts

- Not necessarily the simplest “bitter lesson” way



slido

Please download and install the Slido app on all computers you use



Audience Q&A

ⓘ Start presenting to display the audience questions on this slide.

Outline

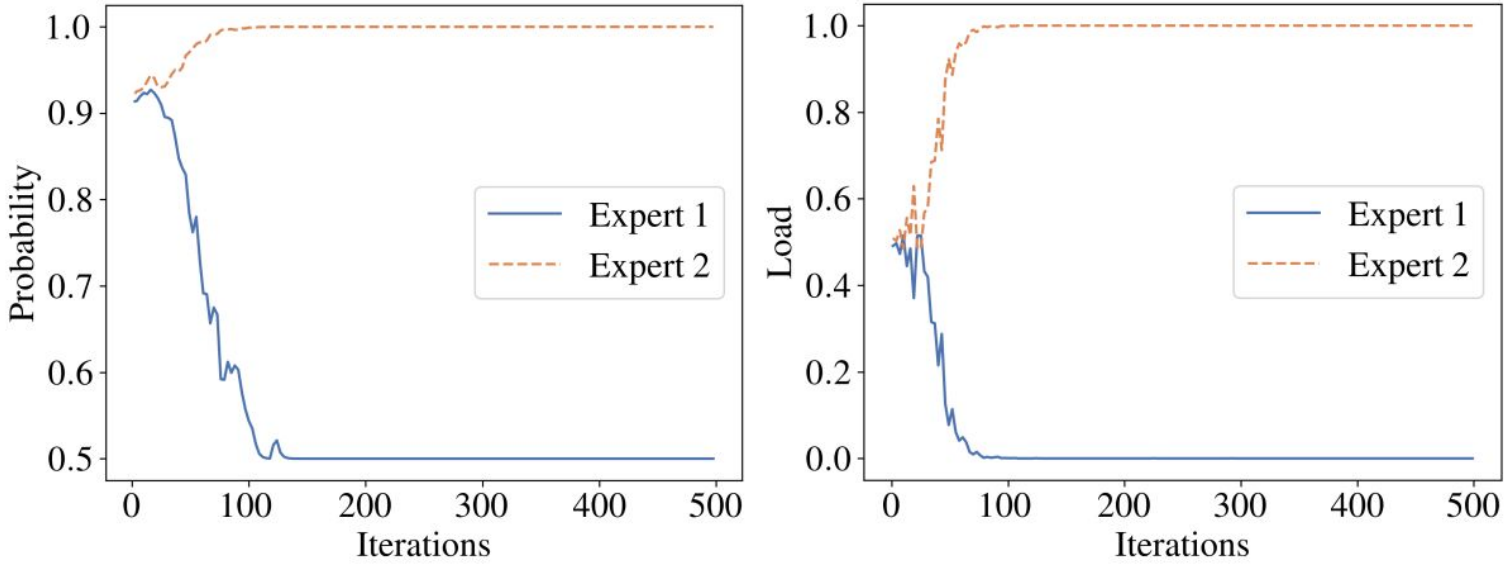
Mixture-of-Experts (MoE)

- Motivation
- Standard Configuration
- Expert Routing
- **Expert Specialization**
- Remarks

Are Experts Really Experts?

Expert routing naturally degenerates

- Does this mean the model does not want to learn experts?

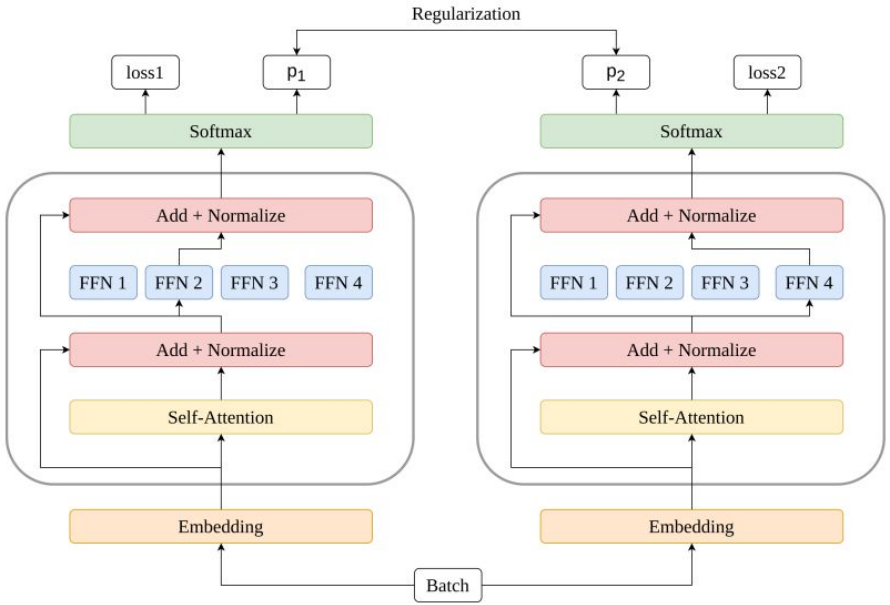


Expert Degeneration in Two Expert Language Models [5]

[5] Zuo et al. 2022. Taming Sparsely Activated Transformer with Stochastic Experts

Random Routing

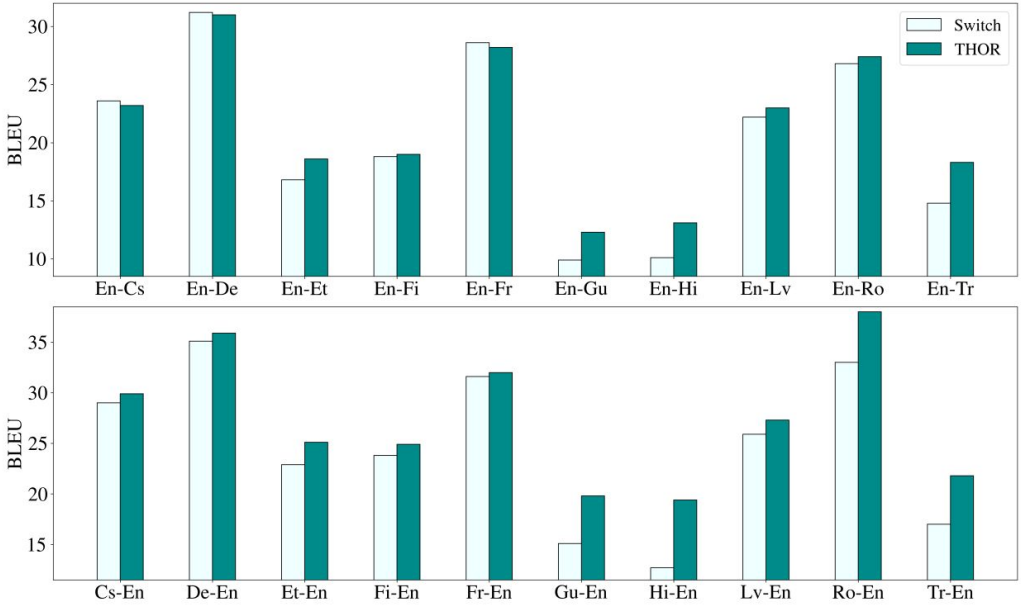
What if we just randomly assign tokens to experts?



Randomly Assign Tokens to Two Experts [5]

Random Routing: Performance

Random routing yields stronger performance than learned gating!

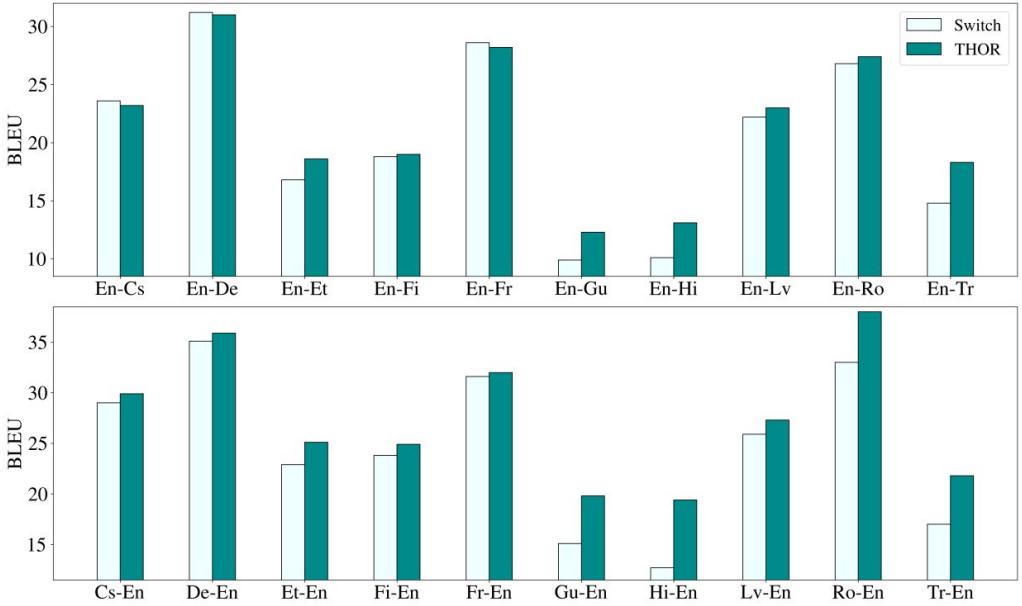


Random (THOR) Routing on Machine Translation Tasks [5]

[5] Zuo et al. 2022. Taming Sparsely Activated Transformer with Stochastic Experts

Random Routing: Performance

Random routing yields stronger performance than learned gating!



Random (THOR) Routing on Machine Translation Tasks [5]

If randomly routed, there is no specialization by design.
More like a FFN level dropout!

[5] Zuo et al. 2022. Taming Sparsely Activated Transformer with Stochastic Experts

Expert Under-Specialization: Why?

There are many attempts to enforce structure within deep neural networks

- Disentangled representations
- Incorporating symbolic information
- Mixture-of-experts

Not much success has observed

- Neural networks do not like these type of inductive biases

Expert Under-Specialization: Why?

There are many attempts to enforce structure within deep neural networks

- Disentangled representations
- Incorporating symbolic information
- Mixture-of-experts

Not much success has observed

- Neural networks do not like these type of inductive biases

What are the challenges for specialized experts?

- Naturally, there are shared knowledge between tokens

Expert Under-Specialization: Why?

There are many attempts to enforce structure within deep neural networks

- Disentangled representations
- Incorporating symbolic information
- Mixture-of-experts

Not much success has observed

- Neural networks do not like these type of inductive biases

What are the challenges for specialized experts?

- Naturally, there are shared knowledge between tokens

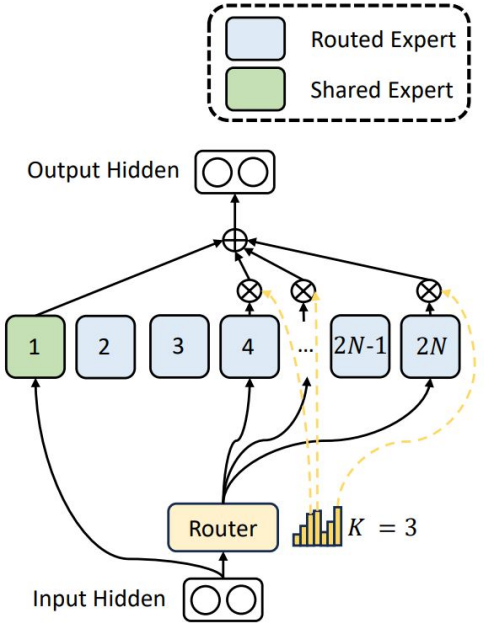
DeepSeekMoE: Shared Experts

Adding shared experts in the mix

- All tokens go through the shared experts + its routed experts

Ideally:

- Common knowledge captured in shared experts
- Routed experts freed up to learn specialty



Adding Shared Experts in MoE [7]

[7] Dai et al. 2024. DeepSeekMoE: Towards Ultimate Expert Specialization in Mixture-of-Experts Language Models

DeepSeekMoE: Shared Experts

Adding shared experts in the mix

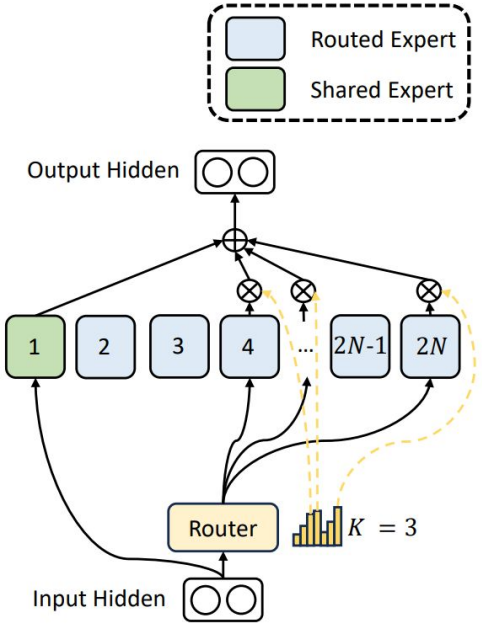
- All tokens go through the shared experts + its routed experts

Ideally:

- Common knowledge captured in shared experts
- Routed experts freed up to learn specialty

No explicit control of common versus special

- Only introducing shared experts in architecture



Adding Shared Experts in MoE [7]

DeepSeekMoE: Performance

Generalizes much better than no shared experts

Metric	# Shot	GShard×1.5	Dense×16	DeepSeekMoE
Relative Expert Size	N/A	1.5	1	0.25
# Experts	N/A	0 + 16	16 + 0	1 + 63
# Activated Experts	N/A	0 + 2	16 + 0	1 + 7
# Total Expert Params	N/A	2.83B	1.89B	1.89B
# Activated Expert Params	N/A	0.35B	1.89B	0.24B
FLOPs per 2K Tokens	N/A	5.8T	24.6T	4.3T
# Training Tokens	N/A	100B	100B	100B
Pile (Loss)	N/A	1.808	1.806	1.808
HellaSwag (Acc.)	0-shot	54.4	55.1	54.8
PIQA (Acc.)	0-shot	71.1	71.9	72.3
ARC-easy (Acc.)	0-shot	47.3	51.9	49.4
ARC-challenge (Acc.)	0-shot	34.1	33.8	34.3
RACE-middle (Acc.)	5-shot	46.4	46.3	44.0
RACE-high (Acc.)	5-shot	32.4	33.0	31.7
HumanEval (Pass@1)	0-shot	3.0	4.3	4.9
MBPP (Pass@1)	3-shot	2.6	2.2	2.2
TriviaQA (EM)	5-shot	15.7	16.5	16.6
NaturalQuestions (EM)	5-shot	4.7	6.3	5.7

Performance on Language Tasks [7]

[7] Dai et al. 2024. DeepSeekMoE: Towards Ultimate Expert Specialization in Mixture-of-Experts Language Models

DeepSeekMoE: Performance

Almost achieving similar performance with dense model

Metric	# Shot	GShard×1.5	Dense×16	DeepSeekMoE
Relative Expert Size	N/A	1.5	1	0.25
# Experts	N/A	0 + 16	16 + 0	1 + 63
# Activated Experts	N/A	0 + 2	16 + 0	1 + 7
# Total Expert Params	N/A	2.83B	1.89B	1.89B
# Activated Expert Params	N/A	0.35B	1.89B	0.24B
FLOPs per 2K Tokens	N/A	5.8T	24.6T	4.3T
# Training Tokens	N/A	100B	100B	100B
Pile (Loss)	N/A	1.808	1.806	1.808
HellaSwag (Acc.)	0-shot	54.4	55.1	54.8
PIQA (Acc.)	0-shot	71.1	71.9	72.3
ARC-easy (Acc.)	0-shot	47.3	51.9	49.4
ARC-challenge (Acc.)	0-shot	34.1	33.8	34.3
RACE-middle (A			46.3	44.0
RACE-high (Acc			33.0	31.7
HumanEval (Pa			4.3	4.9
MBPP (Pass@1)			2.2	2.2
TriviaQA (EM)			16.5	16.6
NaturalQuestio			6.3	5.7

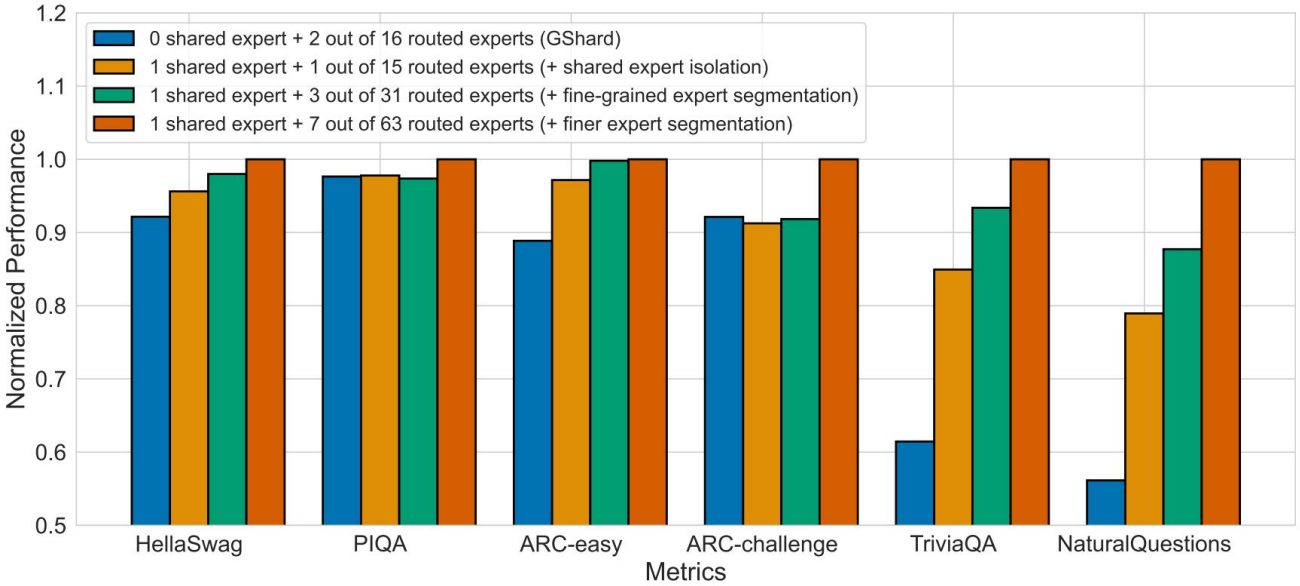
If we view MoE as making a dense model sparse, this is the upper bound?

Performance on Language Tasks [7]

[7] Dai et al. 2024. DeepSeekMoE: Towards Ultimate Expert Specialization in Mixture-of-Experts Language Models

DeepSeekMoE: Performance

Significant benefit from shared expert



DeepSeekMoE: Expert Specialization?

How do we know if the experts are specialized?

If we remove the top routed experts per token, how much worse the model become?

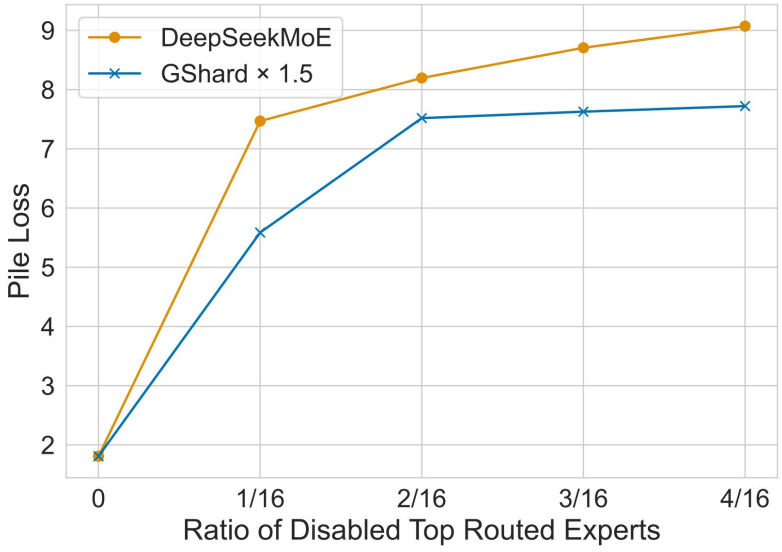
- If experts are not specialized, other experts capture similar information, thus model performance not impact as much
- Otherwise, experts are more specialized

DeepSeekMoE: Expert Specialization?

How do we know if the experts are specialized?

If we remove the top routed experts per token, how much worse the model become?

- If experts are not specialized, other experts capture similar information, thus model performance not impact as much
- Otherwise, experts are more specialized



Relative Loss when top experts are disabled.
↑: expert more specialized [7]

DeepSeekMoE: Alignment

Empirical benefits hold after SFT alignment

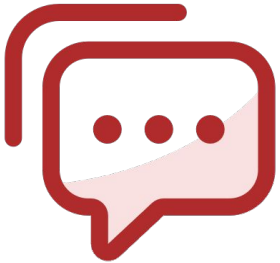
Metric	# Shot	LLaMA2 SFT 7B	DeepSeek Chat 7B	DeepSeekMoE Chat 16B
# Total Params	N/A	6.7B	6.9B	16.4B
# Activated Params	N/A	6.7B	6.9B	2.8B
FLOPs per 4K Tokens	N/A	187.9T	183.5T	74.4T
HellaSwag (Acc.)	0-shot	67.9	71.0	**72.2**
PIQA (Acc.)	0-shot	76.9	78.4	**79.7**
ARC-easy (Acc.)	0-shot	69.7	**70.2**	**69.9**
ARC-challenge (Acc.)	0-shot	**50.8**	50.2	50.0
BBH (EM)	3-shot	39.3	**43.1**	42.2
RACE-middle (Acc.)	5-shot	63.9	**66.1**	64.8
RACE-high (Acc.)	5-shot	49.6	**50.8**	**50.6**
DROP (EM)	1-shot	40.0	**41.7**	33.8
GSM8K (EM)	0-shot	**63.4**	62.6	62.2
MATH (EM)	4-shot	13.5	14.7	**15.2**
HumanEval (Pass@1)	0-shot	35.4	45.1	**45.7**
MBPP (Pass@1)	3-shot	27.8	39.0	**46.2**
TriviaQA (EM)	5-shot	60.1	59.5	**63.3**
NaturalQuestions (EM)	0-shot	**35.2**	32.7	**35.1**
MMLU (Acc.)	0-shot	**50.0**	49.7	47.2
WinoGrande (Acc.)	0-shot	65.1	68.4	**69.0**
CLUEWSC (EM)	5-shot	48.4	66.2	**68.2**
CEval (Acc.)	0-shot	35.1	**44.7**	40.0
CMMLU (Acc.)	0-shot	36.9	**51.2**	49.3

Performance after SFT Alignment [7]

[7] Dai et al. 2024. DeepSeekMoE: Towards Ultimate Expert Specialization in Mixture-of-Experts Language Models

slido

Please download and install the Slido app on all computers you use



Audience Q&A

ⓘ Start presenting to display the audience questions on this slide.

Outline

Mixture-of-Experts (MoE)

- Motivation
- Standard Configuration
- Expert Routing
- Expert Specialization
- **Remarks**

Nested Models

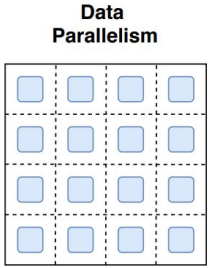
- Nested Embeddings
- Nested Transformers

Hand-in-hand with Infrastructure

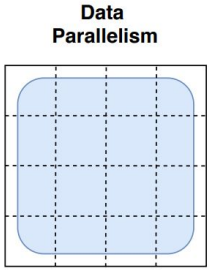
Expert provides another dim to parallel

- When lots of GPUs and perfectly balanced loads, then it comes nearly for free

How the *model weights* are split over cores



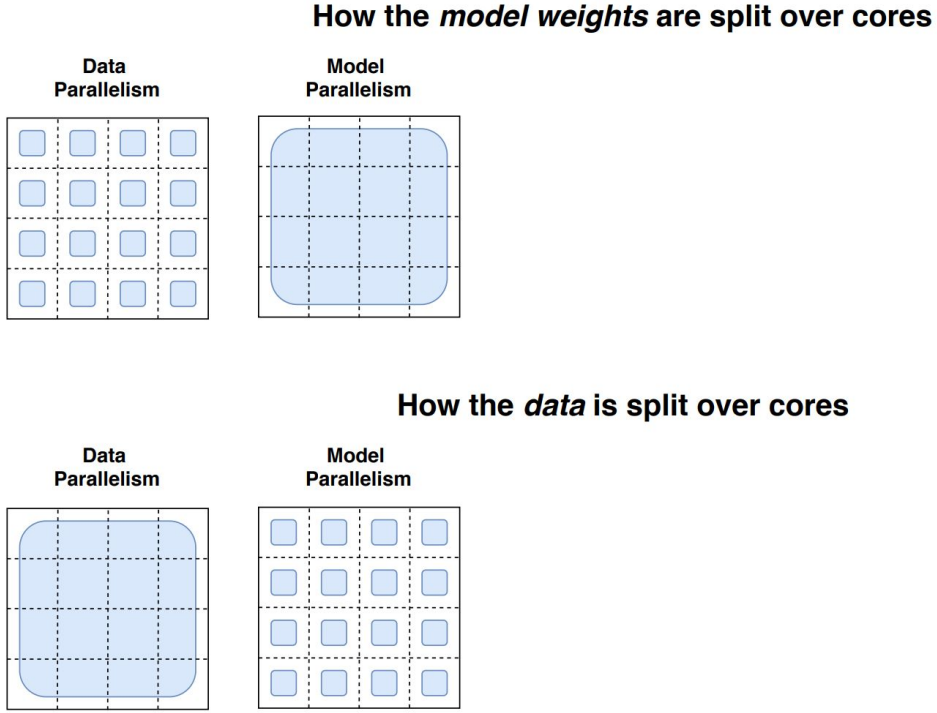
How the *data* is split over cores



Hand-in-hand with Infrastructure

Expert provides another dim to parallel

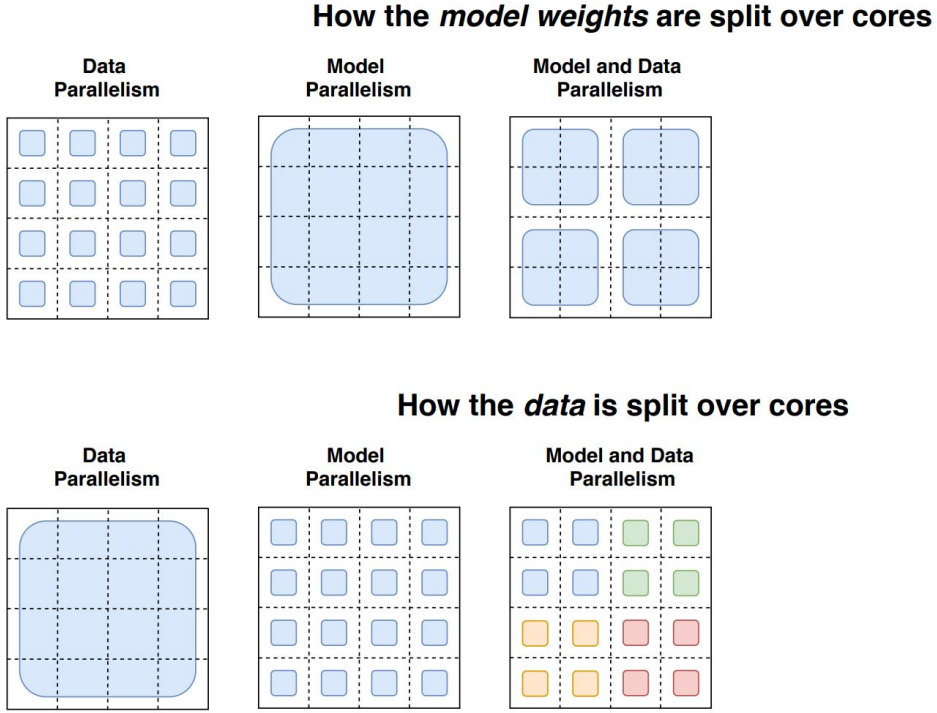
- When lots of GPUs and perfectly balanced loads, then it comes nearly for free



Hand-in-hand with Infrastructure

Expert provides another dim to parallel

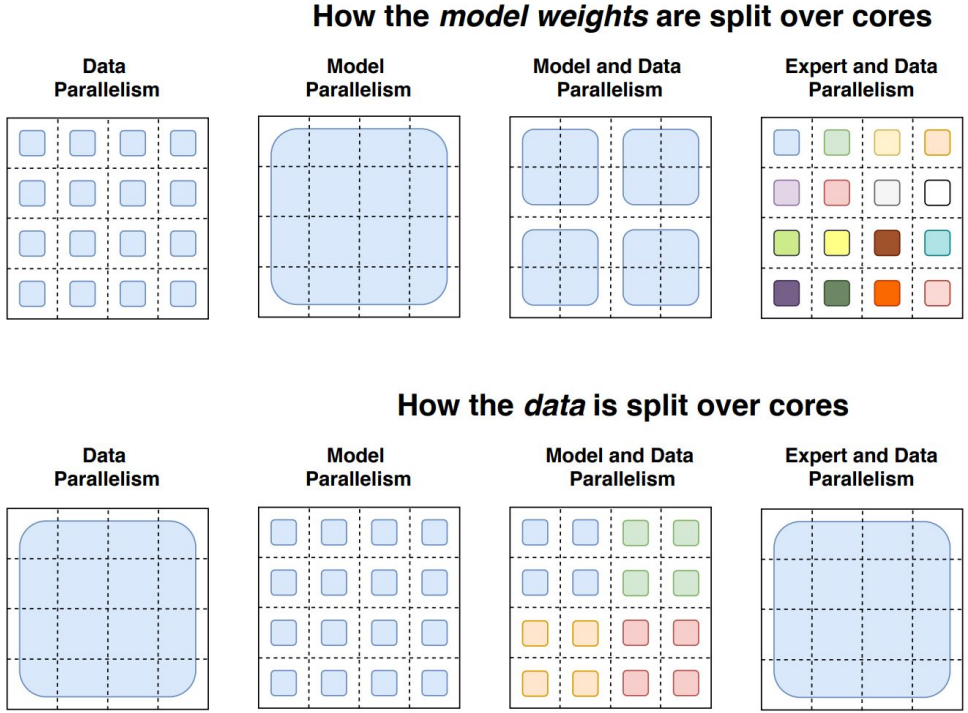
- When lots of GPUs and perfectly balanced loads, then it comes nearly for free



Hand-in-hand with Infrastructure

Expert provides another dim to parallel

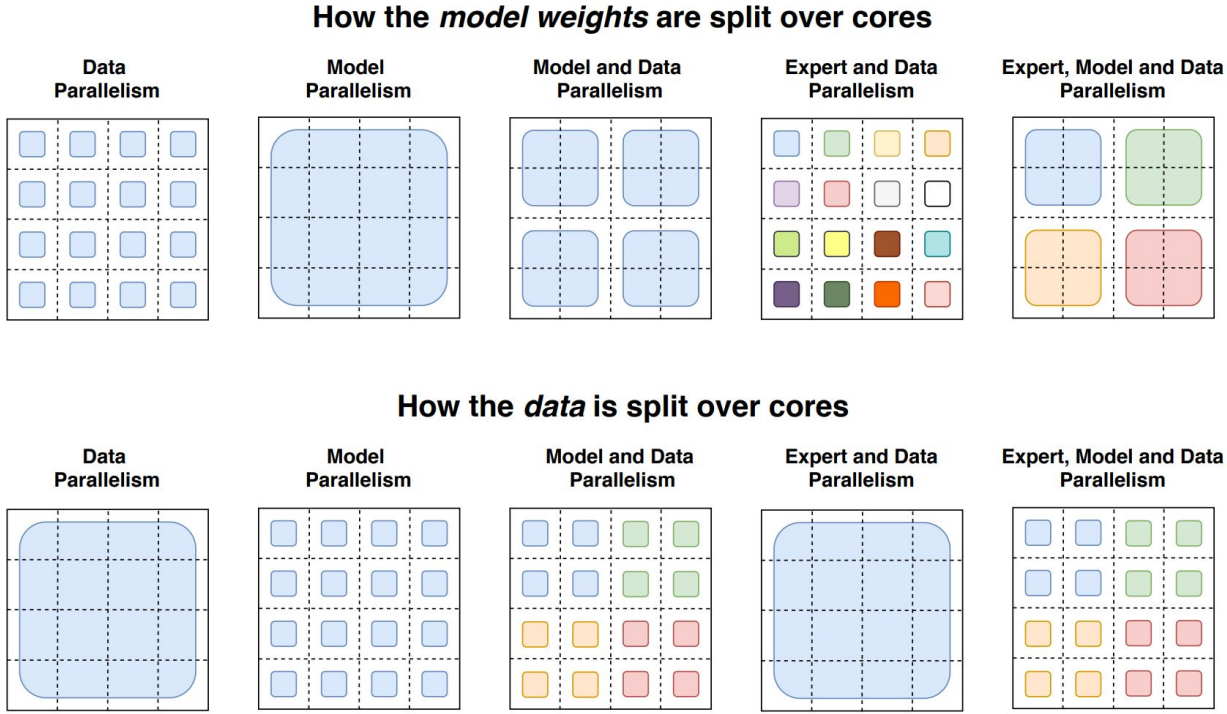
- When lots of GPUs and perfectly balanced loads, then it comes nearly for free



Hand-in-hand with Infrastructure

Expert provides another dim to parallel

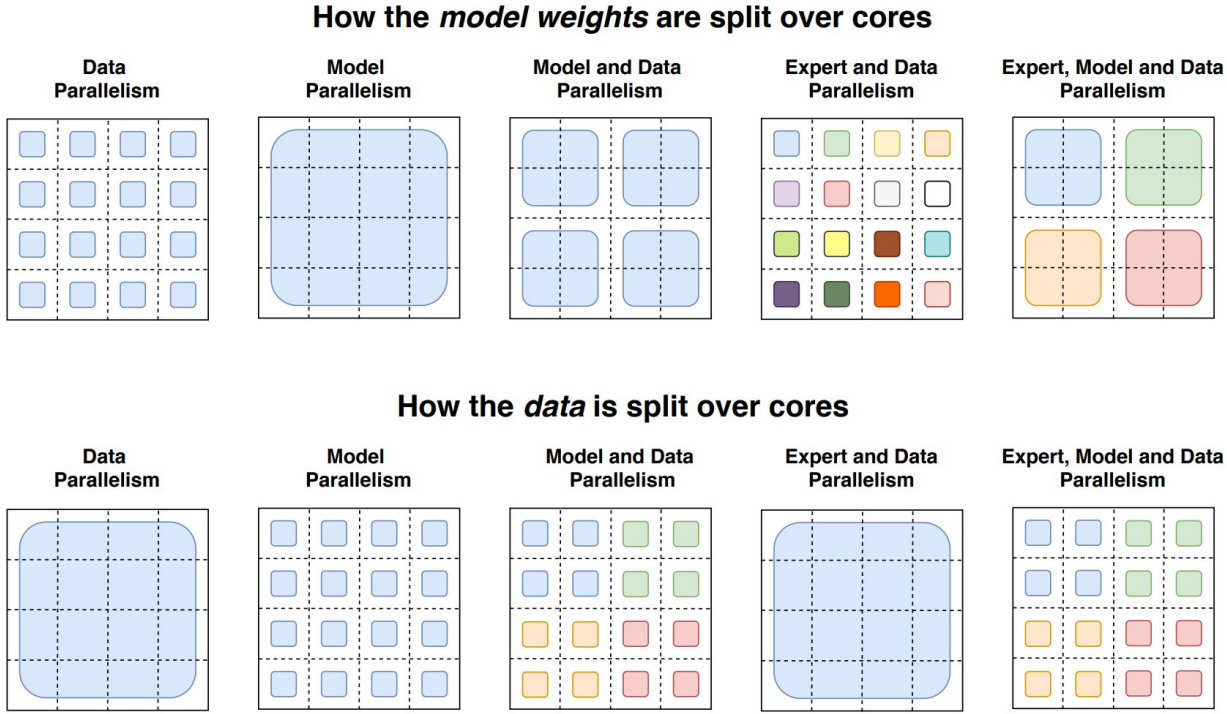
- When lots of GPUs and perfectly balanced loads, then it comes nearly for free



Hand-in-hand with Infrastructure

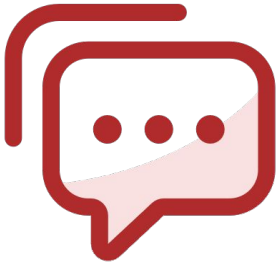
Single Device would not utilize its sparsity well

- Parameters still needs to be kept in GPU memory



slido

Please download and install the Slido app on all computers you use



Audience Q&A

ⓘ Start presenting to display the audience questions on this slide.