Mixture-of-Experts 言語モデルの最適な構造に対する系統的な調査

リュウ　センペイ [1]　新里　顕大[1]　董 于洋 [1]
[1]SB Intuitions 株式会社
{sengpei.liew,kenta.shinzato,yuyang.dong}@sbintuitions.co.jp

概要

現代の混合エキスパート（Mixture-of-Experts, MoE）言語モデルは、総パラメータ数（メモリ使用量）とアクティブパラメータ数（推論コスト）に基づいて設計されている。しかし、これら 2 つの要因だけでは最適なアーキテクチャを記述するには不十分であることがわかった。体系的な研究を通じて、MoE の性能は主に総パラメータ数（N_{total}）とエキスパートの疎性（$s := n_{exp}/n_{topk}$）によって決定されることを示す。ただし、性能は疎性比 s のみで簡潔に記述できるわけではない。なぜならば、エキスパートの総数（n_{exp}）を増やすと、メモリ制約を満たすためにモデルの基幹次元（深さと幅）を縮小せざるを得なくなり、結果として性能にわずかな悪影響（ペナルティ）を与えるためである。本研究では、与えられた制約下で N_{total} を最大化しつつ、s と n_{exp} を最小化するという、MoE 設計の単純な原則を提案する。

1　はじめに

Qwen3-235B-A22B [18] や OLMoE-1B-7B [13] など、公開されている多くの MoE モデルは、総パラメータ数とアクティブパラメータ数で特徴付けられる。これは、総パラメータ数がメモリ使用量を決定し、アクティブパラメータ数が推論時の計算コストを左右するという、実運用上における考慮事項によるものである。

しかし、これら 2 つの指標だけでは MoE アーキテクチャを完全に規定することはできない。我々は MoE アーキテクチャを、層の数（深さ l）、隠れ層の次元（幅 d）、エキスパート数（n_{exp}）、活性化エキスパート数（n_{topk}）、および粒度（$g := d/d_{exp}$）の 5 つの変数で定義する。全結合層 (FFN 層) を全て MoE 層に置き換えた Transformer において、非埋め込みパラメータ数は以下の近似式で表される：

$$N_{total} \approx ld^2\left(4 + 3n_{exp}/g\right) \quad (1)$$

$$N_{active} \approx ld^2\left(4 + 3n_{topk}/g\right) \quad (2)$$

N_{total} と N_{active} を固定しても、幅、深さ、および MoE 構成の選択肢は複数存在する。ここで重要な問いが生じる：「メモリと推論コストの限界が与えられたとき、性能を最適化するためにこれらの変数をどのように選択すべきか？」

本研究では、メモリと推論コストの予算が固定されているという現実的な制約下において、どのようにこれらの変数を組み合わせれば最高の性能を得られるのかという、アーキテクチャ設計の曖昧さを解消するための体系的な調査を行った。

幅と深さの比率および粒度が適切な範囲にあれば、性能は主に N_{total} とエキスパートの疎性 $s := n_{exp}/n_{topk}$ によって決まることを示す。また、特定の n_{exp} ペナルティを特定した。これは、同じ疎性比であっても n_{exp} を増やすと、予算内に収めるために l や d を削る必要があり、トークンあたりの計算能力が実質的に減少するためである。

関連研究　MoE モデルの多角的な側面については、先行研究 [16, 10, 5] において幅広く調査されている。[2] では、MoE モデルの包括的研究が実施されている。本稿では、スケーリング則（scaling laws）を用いて MoE モデルの損失や性能をモデル構成に関連付けている、最も関連性の高い研究に絞って概説する。[3] は、MoE モデルのスケーリング則を提唱した先駆的な研究の一つであり、損失が総パラメータ数 N_{total} とエキスパート数 n_{exp} の関数であることを示した。[9, 11, 1, 12, 17] らは、粒度（granularity）、アップサイクリング、疎性（sparsity）、メモリ効率、および Dense モデルに対する MoE の効率比などの側面を研究している。これらの研究間で結論に細かな違いはあるものの、総パラメータ数およびアクティブパラメータ数が MoE モデルの性

能に影響を与える最も重要な因子であるという点では一致している。しかし、総パラメータ数とアクティブパラメータ数が与えられた条件下で、式 1 および 2 に現れるすべてのパラメータ間の相互作用を解き明かそうとした研究は存在しない。

実験設定　本研究では、Qwen3 [18] シリーズの MoE モデルと類似したアーキテクチャを使用する。これは [5] の設計を継承しており、Transformer ベースの言語モデルの全結合層（FFN 層）を n_{exp} 個のエキスパートへ置換し、各入力トークンに対してそのうちの n_{topk} 個のみを活性化させる構成である。さらに、FFN の隠れ層次元をアテンションの隠れ層次元よりも g 倍小さく設定する「細粒度 MoE（fine-grained MoE）」設計 [4, 9] を採用した。モデルの学習データには FineWeb-Edu [14] を用い、標準的な言語モデル学習の目的関数に従って訓練を行い、ホールドアウトされたデータセットで性能評価を実施した。

2　パラメータ固定による Ablation

N_{total} と N_{active} をほぼ一定に保ちながら、他の変数を変化させて損失（Loss）への影響を調査した。

2.1　Granularity

粒度 g の影響を調べた結果、g が 4 から 8 の範囲で損失が最小化され、それ以上に g を大きくしても収益が逓減することがわかった。先行研究とも一致し、本実験では $g = 4$ を採用する。

l	d	g	n_{exp}	n_{topk}	Loss diff (%)
		2	64	4	0.7%
		4	128	8	0.48%
8	384	**8**	**256**	**16**	**0.00%**
		16	512	32	0.21%
		2	64	4	1.04%
		4	**128**	**8**	**0.00%**
18	1024	8	256	16	0.29%
		16	512	32	0.58%

表 1　Ablation study on granularity g with fixed total and active parameters.

2.2　幅と深さの比率 (Width-to-depth Ratio)

比率 $\gamma := d/l$ を変化させた実験では、γ が極端に小さい（例：10）と損失が高くなり、中間的な値（例：42）で最も低い損失が得られた。標準的な設定 [8] に従い、γ を 32 から 64 の間に保つことが推奨される。

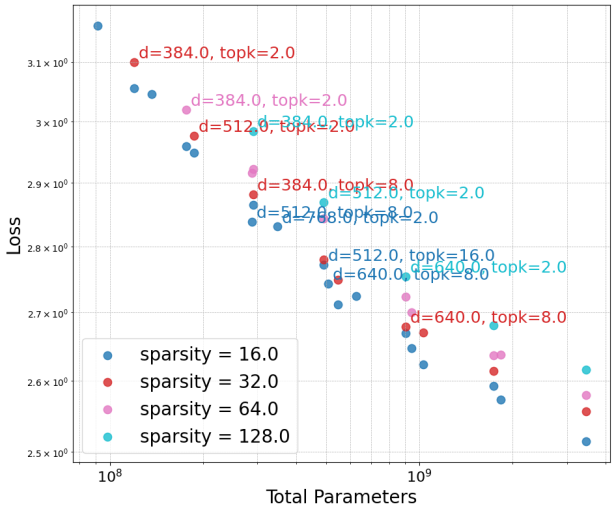

図 1　Loss vs. Total Parameters. Each point represents a model with a specific configuration.

l	d	d/l	n_{nexp}	n_{topk}	Loss diff (%)
16	240	15	43	4	0.65%
8	**336**	**42**	**43**	**4**	**0.00%**
4	480	120	43	4	1.12%
16	160	10	103	16	2.68%
8	224	28	103	16	2.20%
4	320	80	103	16	1.68%

表 2　Ablation study on the width-to-depth ratio $\gamma = d/l$ with approximately the same total and active parameters.

3　MoE 設計のスケーリング則

30M から 3B パラメータのモデルを用いて、様々な構成でスケーリング則をフィッティングした。我々は、損失 L と各変数の間にべき乗則の関係を仮定し、対数スケールで線形回帰を行った (図 3)。

スケーリング則のフィッティングにあたっては、すべての変数の対数を取り、最小二乗法（OLS）を用いて線形回帰を行った。フィッティングおよび統計的検定には statsmodels ライブラリ [15] を使用している。各変数の有意性検定は、有意水準 0.05 の t 検定によって実施した。

以下のことが言える。

1. N_{total} は単一の指標として最も強力な予測因子である。
2. N_{total} に s を加えることで、適合度（R^2）が大きく向上する。s は MoE モデルの性能の主要な調整弁の 1 つである。
3. 主要なスケーリング則として

$(N_{total}, n_{exp}, n_{topk})$ の採用をした 4 式を提案する。

$$L \propto {N_{total}}^{-0.052} {n_{exp}}^{0.023} {n_{topk}}^{-0.018} \quad (3)$$

$$= {N_{total}}^{-0.052} {s}^{0.018} {n_{exp}}^{0.005} \quad (4)$$

この式は、疎性 s が低いほど（つまりより多くのエキスパートを活性化するほど）性能が向上することを示している。また、n_{exp} の指数が正であることは、同じ総パラメータ数であれば、エキスパート数を不必要に細分化するよりも、基幹モデルの次元 (l, d) を大きく保つ方が有利であることを示唆している。ただし、実用的な設計においては、式 1 および 2 に含まれる各パラメータが整数でなければならないという事実によって、さらなる制約が生じることに注意が必要です。さらに、d（モデルの次元数）は、アテンションヘッドの分割や、（テンソル並列化を使用する場合の）並列化の要件に基づくパーティショニング（整除性）の制約を満たさなければならない。

その結果、我々は以下の MoE モデルの設計ルーチンを提案する 1。まず、メモリ容量、計算グラフ (g)、および並列化のためのパーティショニング制約の範囲内で、最大となる (l, d)（層数と次元数）の組み合わせを算出する。これを γ（容量係数）と n_{exp}（エキスパート数）の各パターンに対して繰り返し実行し、N_{total}（総パラメータ数）を最大化します。次に、推論コストの予算が飽和するまで n_{topk} を最大化する。この設計ルーチン 1 は、過剰な n_{exp} によって生じる性能上のペナルティを防ぎつつ、利用可能なメモリを最大限に活用することを保証するものである。

Qwen3-235B-A22B との比較　提案した最適化原則を用い、Qwen3-235B-A22B と同様のメモリおよび推論制約下における MoE 構成を導出した。Qwen3 のメモリおよび推論制約を適用した結果、我々の最適化ルーチンは $(l, d, n_{exp}, n_{topk}) = (94, 4096, 128, 7)$ という MoE 構成を提示した。これにより、総パラメータ数は 234B、アクティブパラメータ数は 21.7B となる。

この構成は、実際に $(l, d, n_{exp}, n_{topk}) = (83, 5312, 128, 8)$ を採用している Qwen3-235B-A22B に近いものであるが、我々の導出結果の方がより多くの層数（深さ）を持つ。また、Qwen3-235B-A22B においては $g = 2.7$ が使用されている点も併記しておく。

Algorithm 1 MoE Architectural Optimization

Input: Memory constraint C_{total}, Inference constraint C_{active}, Alignment factor k_{align}
Parameters: $\Gamma \in [32, 64]$, $n_{exp} \in \{2^1, 2^2, \ldots, 2^k\}$
Initialize: $L_{min} \leftarrow \infty, \theta^* \leftarrow \emptyset$
for each $n_{exp} \in \{2^1, \ldots, 2^k\}$ **do**
  for each $\gamma \in \Gamma$ **do**
    $l \leftarrow \lfloor (C_{total}/(\gamma^2(4 + 0.75 n_{exp})))^{1/3} \rfloor$
    $d \leftarrow \text{round}(\gamma \cdot l / k_{align}) \cdot k_{align}$
    while $l \cdot d^2 \cdot (4 + 0.75 n_{exp}) > C_{total}$ **do**
      $d \leftarrow d - k_{align}$
    end while
    $n_{topk} \leftarrow \min\left(n_{exp}, \lfloor \frac{4}{3}(\frac{C_{active}}{ld^2} - 4) \rfloor\right)$
    if $n_{topk} \geq 1$ **then**
      $L \leftarrow (ld^2(4+0.75n_{exp}))^{-0.052} \cdot n_{exp}^{0.023} \cdot n_{topk}^{-0.018}$
      if $L < L_{min}$ **then**
        $L_{min} \leftarrow L, \theta^* \leftarrow (l, d, n_{exp}, n_{topk})$
      end if
    end if
  end for
end for
return θ^*

Variables	R^2	Result/Intrepretation
Simple Baseline		
$\log(N_{total})$ only	0.926	Baseline (Total params).
$\log(N_{active})$ only	0.641	Baseline (Active params).
Core Functions		
$\log(N_{active}) + \log(s)$	0.944	Moderately good fit.
$\log(N_{total}) + \log(s)$	0.983	Good fit.
$\log(N_{total}) + \log(n_{exp}) + \log(n_{topk})$	**0.985**	**Good and disambiguated fit.**
$\log(N_{active}) + \log(n_{exp}) + \log(n_{topk})$	0.981	Moderately good fit.
Interaction Functions		
$\log(N_{total}) + \log(s) + \log(N_{total})\log(s)$	0.983	Interaction term redundant.
$\log(N_{total}) + \log(N_{active}) + \log(N_{total})\log(N_{active})$	0.988	Strong multicollinearity problem.
Other Combinations		
$\log(N_{total}) + \log(N_{active}) + \log(s)$	0.985	N_{active}'s significance is low.

表 3 Results of fitting various combinations of variables to loss. More results in Table **??** of Appendix **??**.

3.1 設計の検証

式 4 におけるスケーリング則は、固定されたトークン予算で学習されたモデルに基づいている。MoE 設計の指針としての有効性を検証するため、計算リソースの制約から 2 つの構成を固定した上で、様々なモデルサイズおよびデータセットサイズでモデルを学習させる実験を行った。

具体的には、同じ疎性 s を持つ MoE 構成、すなわち $(n_{exp}, n_{topk}) = (128, 8)$ および $(256, 16)$ を比較した。これらの構成に対して、以下の通り Chinchilla 形式のスケーリング則 [6] をフィッティングした。

$$L_{128/8 \text{ or } 256/16} = AN_{total}^{-\alpha} + BD^{-\beta} + E \tag{5}$$

ここで、D はデータセットサイズ（トークン数）であり、A, B, E, α, β はフィッティングされた係数である。式 4 に基づけば、同じ N_{total}、s、および D の下で比較した場合、$(128, 8)$ 構成は n_{exp} がより小さいため、$(256, 16)$ 構成よりも優れた性能を示すはずである。

我々は実験を実施し、それに応じてスケーリング則のフィッティングを行った。得られた曲線は図 2 に示す通りである。実際に、同じ N_{total} および D において $(128, 8)$ 構成は一貫して $(256, 16)$ 構成を上回っており、我々の設計原理の有効性が確認された。

先行研究 [1, 17] においても、モデル構成とデータセットサイズを含むスケーリング則のフィッティングが試みられている。しかし、それらは s のみを対象としており、n_{exp} と n_{topk} を分離してはいなかった。我々の結果は、これらすべての要因を含む、より完全なスケーリング則が望ましいことを示唆している。

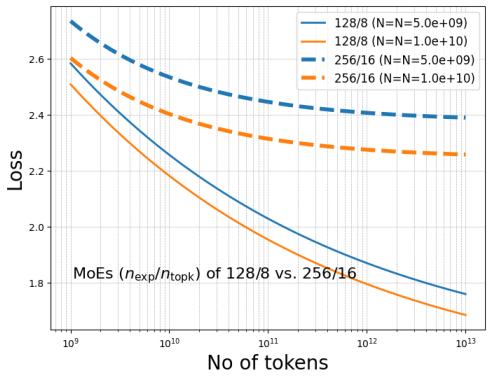

図 2 Comparison of loss between the $(128, 8)$ and $(256, 16)$ configurations with fitted scaling laws.

4 終わりに

結論 MoE モデルは大規模言語モデルにおいて主流となっているが [7, 4, 18]、その最適な構成については依然として理解が進んでいなかった。本研究では、MoE の構成要素を分離することで、N_{total}（総パラメータ数）が性能の主要な予測因子であり、次いでエキスパートの疎性 s、および n_{exp}（エキスパート総数）が重要であることを示した。実用的な制約条件下でこれらの数値を最適化するという我々の設計原理は、より効率的な MoE モデルを構築するための有用な基盤を提供するものである。

参考文献

[1] Samira Abnar, Harshay Shah, Dan Busbridge, Alaaeldin Mohamed Elnouby Ali, Josh Susskind, and Vimal Thilak. Parameters vs flops: Scaling laws for optimal sparsity for mixture-of-experts language models. **arXiv preprint arXiv:2501.12370**, 2025.

[2] Weilin Cai, Juyong Jiang, Fan Wang, Jing Tang, Sunghun Kim, and Jiayi Huang. A survey on mixture of experts. **arXiv preprint arXiv:2407.06204**, 2024.

[3] Aidan Clark, Diego de Las Casas, Aurelia Guy, Arthur Mensch, Michela Paganini, Jordan Hoffmann, Bogdan Damoc, Blake Hechtman, Trevor Cai, Sebastian Borgeaud, et al. Unified scaling laws for routed language models. pages 4057–4086. PMLR, 2022.

[4] Damai Dai, Chengqi Deng, Chenggang Zhao, RX Xu, Huazuo Gao, Deli Chen, Jiashi Li, Wangding Zeng, Xingkai Yu, Y Wu, et al. Deepseekmoe: Towards ultimate expert specialization in mixture-of-experts language models. **arXiv preprint arXiv:2401.06066**, 2024.

[5] William Fedus, Barret Zoph, and Noam Shazeer. Switch transformers: Scaling to trillion parameter models with simple and efficient sparsity. **Journal of Machine Learning Research**, 23(120):1–39, 2022.

[6] Jordan Hoffmann, Sebastian Borgeaud, Arthur Mensch, Elena Buchatskaya, Trevor Cai, Eliza Rutherford, Diego de Las Casas, Lisa Anne Hendricks, Johannes Welbl, Aidan Clark, et al. Training compute-optimal large language models. **Proceedings of the 36th International Conference on Neural Information Processing Systems**, pages 30016–30030, 2022.

[7] Albert Q Jiang, Alexandre Sablayrolles, Antoine Roux, Arthur Mensch, Blanche Savary, Chris Bamford, Devendra Singh Chaplot, Diego de las Casas, Emma Bou Hanna, Florian Bressand, et al. Mixtral of experts. **arXiv preprint arXiv:2401.04088**, 2024.

[8] Jared Kaplan, Sam McCandlish, Tom Henighan, Tom B Brown, Benjamin Chess, Rewon Child, Scott Gray, Alec Radford, Jeffrey Wu, and Dario Amodei. Scaling laws for neural language models. **arXiv preprint arXiv:2001.08361**, 2020.

[9] Jakub Krajewski, Jan Ludziejewski, Kamil Adamczewski, Maciej Pióro, Michał Krutul, Szymon Antoniak, Kamil Ciebiera, Krystian Król, Tomasz Odrzygóźdź, Piotr Sankowski, et al. Scaling laws for fine-grained mixture of experts. **arXiv preprint arXiv:2402.07871**, 2024.

[10] Dmitry Lepikhin, HyoukJoong Lee, Yuanzhong Xu, Dehao Chen, Orhan Firat, Yanping Huang, Maxim Krikun, Noam Shazeer, and Zhifeng Chen. Gshard: Scaling giant models with conditional computation and automatic sharding.

[11] Seng Pei Liew, Takuya Kato, and Sho Takase. Scaling laws for upcycling mixture-of-experts language models. In Aarti Singh, Maryam Fazel, Daniel Hsu, Simon Lacoste-Julien, Felix Berkenkamp, Tegan Maharaj, Kiri Wagstaff, and Jerry Zhu, editors, **Proceedings of the 42nd International Conference on Machine Learning**, volume 267 of **Proceedings of Machine Learning Research**, pages 37682–37704. PMLR, 13–19 Jul 2025.

[12] Jan Ludziejewski, Maciej Pióro, Jakub Krajewski, Maciej Stefaniak, Michał Krutul, Jan Małaśnicki, Marek Cygan, Piotr Sankowski, Kamil Adamczewski, Piotr Miłoś, et al. Joint moe scaling laws: Mixture of experts can be memory efficient. **arXiv preprint arXiv:2502.05172**, 2025.

[13] Niklas Muennighoff, Luca Soldaini, Dirk Groeneveld, Kyle Lo, Jacob Morrison, Sewon Min, Weijia Shi, Pete Walsh, Oyvind Tafjord, Nathan Lambert, et al. Olmoe: Open mixture-of-experts language models. **arXiv preprint arXiv:2409.02060**, 2024.

[14] Guilherme Penedo, Hynek Kydlíček, Anton Lozhkov, Margaret Mitchell, Colin A Raffel, Leandro Von Werra, Thomas Wolf, et al. The fineweb datasets: Decanting the web for the finest text data at scale. **Advances in Neural Information Processing Systems**, 37:30811–30849, 2024.

[15] Skipper Seabold and Josef Perktold. statsmodels: Econometric and statistical modeling with python. In **9th Python in Science Conference**, 2010.

[16] Noam Shazeer, Azalia Mirhoseini, Krzysztof Maziarz, Andy Davis, Quoc Le, Geoffrey Hinton, and Jeff Dean. Outrageously large neural networks: The sparsely-gated mixture-of-experts layer. 2017.

[17] Changxin Tian, Kunlong Chen, Jia Liu, Ziqi Liu, Zhiqiang Zhang, and Jun Zhou. Towards greater leverage: Scaling laws for efficient mixture-of-experts language models. **arXiv preprint arXiv:2507.17702**, 2025.

[18] An Yang, Anfeng Li, Baosong Yang, Beichen Zhang, Binyuan Hui, Bo Zheng, Bowen Yu, Chang Gao, Chengen Huang, Chenxu Lv, et al. Qwen3 technical report. **arXiv preprint arXiv:2505.09388**, 2025.