# TopK Language Models

高橋 良允 [1,3]　稲葉 達郎 [2]　乾 健太郎 [2,1,3]　Benjamin Heinzerling[3,1]
[1] 東北大学　[2]MBZUAI　[3] 理化学研究所
ryosuke.takahashi@dc.tohoku.ac.jp

## 概要

スパースオートエンコーダ（SAE）は，言語モデル（LM）の活性化空間を疎な特徴へ分解し，内部表現の分析・解釈を可能にする有力な手法である．一方で，SAE は事後学習を必要とするため，有用性や内的妥当性を損なう課題が残る．本研究ではこの課題に対処するため，Transformer LM の各層の出力に TopK 活性化関数を組み込んだ TopK LM を提案する．TopK LM は，事後的な SAE 学習を不要としつつ，SAE に類する解釈可能性を実現する新しい基盤モデルである．単純なアーキテクチャ変更にもかかわらず，TopK LM は元来の能力を維持したまま解釈可能性という利点を提供する．これにより，LM が概念をいかに学習し表現するかを理解するための安定で信頼できる基盤となり，LM の解釈可能性に関する研究を前進させることが期待される．

## 1　はじめに

スパースオートエンコーダ (SAE) [1–3]，とりわけ TopK SAE [4] は言語モデル (LM) の内部表現を解釈するための重要な手法である [5–8]．しかし，その有用性および内的妥当性を低下させるいくつかの欠点を抱えている．SAE は事後的に学習され，入力を完全には再構成できないため疎性と再構成忠実度の間にトレードオフが生じる [4, 9]．また事後学習であるがゆえに，たとえば `Golden Gate Bridge` という概念を符号化する特徴 [3] が得られない場合，それが (a) 学習過程において SAE が当該特徴の発見に失敗したためなのか，(b) 基盤となる LM 側に表現が存在しないためなのか [10, 11] の判別が困難である．加えて，事後学習を要することに起因して初期条件への感度も高く，同一の LM の隠れ状態に対して学習したとしても乱数シードが異なる SAE は異なる特徴を学習し得ることが報告されている [12]．これらの欠点は，SAE を用いてモデルの学習ダイナミクスを分析する際に一層深刻となる [13, 14]．

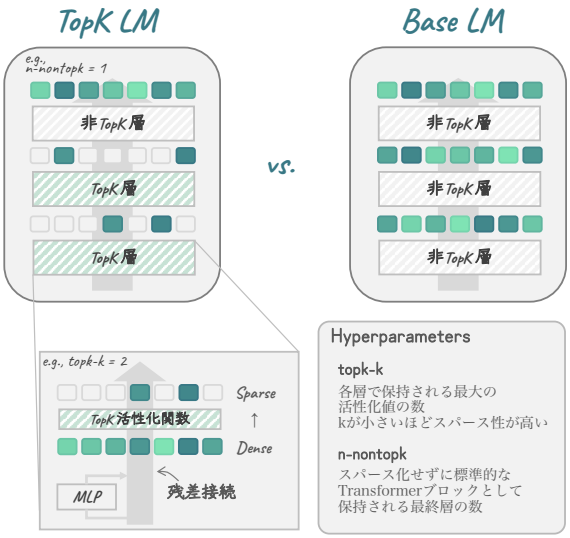


図 1: **TopK LM**：TopK 活性化関数の導入により内在的な解釈可能性を実現した LM．TopK 層では上位 $k$ 個の値のみを選択的に活性化させることで疎性を導入し，最終層では密な処理を維持する．

本研究の目的は，事後学習を必要とせず，内在的に解釈可能な LM アーキテクチャを設計することである．具体的には，Transformer LM の各層の出力活性値において TopK 活性化関数を適用するだけで，SAE 類似の疎性を実現する．TopK 活性化は，単純であるにもかかわらず極めて有効な疎性制約であることが示されている [4]．得られた TopK LM は，密活性化モデルと同程度の性能を維持しつつ，SAE 的な解釈可能性を提供する．

以上より，本研究の貢献は以下のとおりである．

- 疎活性化を用いた新たな Transformer LM アーキテクチャである TopK LM を提案する．
- 高い疎性の下でも TopK LM はベースライン LM に匹敵する性能を保つことを示す．
- TopK LM の解釈可能性をニューロンの単一概念特化性，因果的介入性の観点から提示する．

## 2 TopK Language Model

### 2.1 アーキテクチャ

本節では，選択した層に TopK 活性化関数を挿入することで活性化の疎性を導入する TopK 言語モデル（TopK LM）のアーキテクチャを述べる．その概略は，ベースラインとなる LM との比較として 図 1 に示す．

**TopK 活性化** $x = (x_1, \ldots, x_d) \in \mathbb{R}^d$ とする．また $\{x_1, \ldots, x_d\}$ における $k$ 番目に大きい値を $\tau_k(x)$ と表す．ここで $k$ はハイパーパラメータであり，保持する活性化の個数を指定することで疎性の度合いを制御する．TopK 活性化関数 $\mathcal{T}_k : \mathbb{R}^d \to \mathbb{R}^d$ は，各成分ごとに次のように定義される：

$$y_i = \begin{cases} f(x_i) & x_i \geq \tau_k(x) \\ 0 & \text{otherwise} \end{cases} \tag{1}$$

あるいは同値に，

$$y_i = f(x_i)\,\mathbf{1}_{\{x_i \geq \tau_k(x)\}} \quad i = 1, \ldots, d \tag{2}$$

と書ける．ここで $f$ は要素ごとの非線形関数（例：ReLU）である．

**アニーリング付き TopK 平滑化** 学習安定性および収束性の改善を目的として，学習を通じて疎性の度合いを変化させることを検討した．現行の LM 学習手法が主として密モデル向けに最適化されていることを踏まえると，完全に密な状態から TopK 疎モデルへと段階的に遷移させることが有益である可能性がある．実際，アニーリングを伴う TopK 活性化が良好に機能することを経験的に確認し，この仮説が支持された．具体的には，アニーリング係数 $\alpha \in [0, 1]$ を導入し，学習の進行に伴って $\alpha$ を 1 から 0 へ減衰させることで，活性化を密から疎へと連続的に移行させる．予備実験では線形減衰が良好であったため，本稿ではこれを用い，より洗練された減衰スケジュールの検討は今後の課題とする．線形アニーリングされた TopK 活性化は次式で与えられる：

$$y = \alpha\, f(x) + (1 - \alpha)\big(f(x) \odot \mathbf{1}_{\{x \geq \tau_k(x)\}}\big) \tag{3}$$

ここで $f(x) = (f(x_1), \ldots, f(x_d))$ であり，$\odot$ は要素ごとの積を表す．言い換えると，学習初期（$\alpha = 1$）ではブロックは完全に密であり，$\alpha \to 0$ とともに TopK でない活性化の寄与が小さくなり，$\alpha = 0$ でこの活性化関数は TopK と一致する．本手法は dense-to-sparse 学習 [15, 16] と関連するが，同一ではない．

**ハイブリッドブロック配置** 後半層における表現能力を維持しつつ疎性の利点も得るため，最終層に残す密な Transformer 層の数を表すハイパーパラメータ $n_{\text{nontopk}}$ を導入する．具体的には，全層数を $L$ としたとき，先頭から $L - n_{\text{nontopk}}$ 層を TopK 層に置き換え，残りの最後 $n_{\text{nontopk}}$ 層は非 TopK 層とする．このハイブリッド設計により，アーキテクチャの大部分で疎性を活用しつつ，最終 $n_{\text{nontopk}}$ 層においては十分な表現力を保持できる．

### 2.2 事前学習設定

本節では，§ 2.1 で導入した TopK LM の学習設定について述べる．

**モデル構成** 事前学習は，Llama アーキテクチャ [17] のデコーダのみの構成に基づく 2 種類のモデルで実施する．すなわち，標準的な構成のベースライン LM と，TopK 機構を組み込んだ TopK LM である．両モデルは層数 $L \in \{8, 16, 24\}$ として構成し，ベースライン LM の隠れ次元は $D = 1024$，TopK LM の隠れ次元は $D \in \{2048, 3072\}$ とする．TopK LM の隠れ次元を拡張しているのは，SAE のような TopK 制約を課す条件下でもベースライン LM と同様の表現力を確保するためである．いずれの設定においても注意ヘッド数を $D/128$ とする．また，非 TopK 層数を $n_{\text{nontopk}} = 2$ とし，学習ステップの最初の 20% にわたってアニーリングを適用する．

**トークナイザとデータセット** すべてのモデルは，語彙サイズ 32000 の Llama トークナイザを共有し，入力処理の一貫性を確保する．事前学習データには FineWeb Edu コーパス [18, 19] を用い，学習に用いるデータ量は約 200 億トークンである．同一コーパスから分割した保持データを検証セットとして用いる．

### 2.3 学習後の基本性能評価

LM の事前学習に関する研究 [20, 21] に従い，各事前学習済みモデルをパープレキシティによる言語的流暢性と，下流タスクの正解率による 2 軸から評価する．評価には lm-evaluation-harness [22] を用いる．

第一に，FineWeb-Edu の検証データに対するパープレキシティを報告する．第二に，常識推論および質問応答にまたがる複数選択式ベンチマーク群に対する正解率を測定する．対象には LAMBADA [23],

表 1: 検証セットにおけるパープレキシティ（↓）と，6 つの下流ベンチマーク（LAMBADA，ARC-Easy，ARC-Challenge，WinoGrande，OpenBookQA，HellaSwag）およびそれらの平均（AVG）における正解率（↑）を，層数 $L \in \{8, 16, 24\}$ のベースライン LM および TopK LM モデルについて示す．また，ベースライン LM の隠れ次元は $D = 1024$，TopK LM は $D = 3072$ である．

| $L$ | Perplexity ↓ | | Accuracy ↑ | | | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | Valid. | | LAMBADA | | ARC-e | | ARC-c | | Winogrande | | OBQA | | HellaSwag | | AVG | |
| | Base | TopK | Base | TopK | Base | TopK | Base | TopK | Base | TopK | Base | TopK | Base | TopK | Base | TopK |
| 8 | 2.686 | **2.357** | 16.53 | **18.77** | **47.52** | 38.68 | 24.74 | **27.30** | **52.72** | 50.28 | 29.00 | **30.40** | 31.44 | **32.96** | **33.66** | 33.06 |
| 16 | 2.539 | **2.277** | **20.20** | 18.49 | **54.12** | 44.36 | **28.07** | 26.19 | 49.41 | **50.04** | **31.60** | **31.60** | 34.51 | **35.94** | **36.32** | 34.44 |
| 24 | 2.465 | **2.246** | **21.37** | **21.37** | **57.37** | 47.05 | **29.01** | 27.99 | 49.41 | **51.46** | 31.60 | **34.00** | **37.39** | 36.66 | **37.69** | 36.42 |

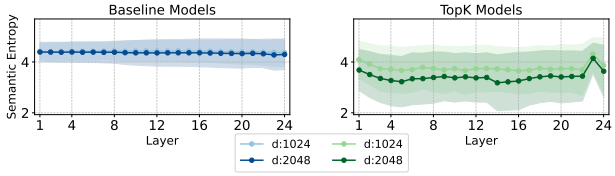


図 2: ベースラインモデル (左) と TopK モデル (右) のセマンティックエントロピーの比較．横軸はモデルの層番号を示す．実線は各層における平均エントロピーを表し、網掛け部分は平均値 ±1 標準偏差の範囲を示す．

WinoGrande [24]，HellaSwag [25]，ARC [26] の Easy および Challenge，ならびに OpenBookQA [27] を含める．これらの評価により，各モデルの生成的流暢性と創発的な推論能力を包括的に把握できる．

表 1 は，隠れ次元 $D = 1024$ のベースライン LM と，より大きい隠れ次元 $D = 3072$ を持つ TopK LM について，層数 $L \in \{8, 16, 24\}$ での検証パープレキシティと，下流 6 ベンチマークの正解率（および平均：AVG）を比較した結果を示す．まず，検証セットにおけるパープレキシティは，全ての層数で TopK LM がベースラインを改善し（例：$L = 24$ で 2.465 → 2.246），TopK 機構を導入しても基本的な言語モデリング性能が損なわれないことが確認できる．

下流タスクについても，TopK LM の正解率は全体としてベースラインと近い水準にあり，AVG の差はおおむね 1 ~ 2 ポイント程度に収まる（$L = 8$: 33.66 vs. 33.06，$L = 16$: 36.32 vs. 34.44，$L = 24$: 37.69 vs. 36.42）．個別には，$L = 8$ では LAMBADA（16.53% → 18.77%）や ARC-c（24.74% → 27.30%），$L = 24$ では WinoGrande（49.41% → 51.46%）および OBQA（31.60% → 34.00%）などで TopK LM がベースラインを上回る一方，ARC-e では一貫してベースラインが優位である．

以上より，TopK による疎活性化制約を課しても，ベースラインと比べて下流タスク性能は概ね同等の範囲に保たれており，尚且つ言語モデリングの性能は上回ることから，TopK LM がベースライン LM と十分に競争力のある性能を示すことが分かる．

# 3 解釈可能性検証

## 3.1 ニューロンの単一概念特化性

24 層 Transformer LM (隠れ状態次元 1024 および 2048）に対し，TopK 活性化（$k = 64$，ただし最終 2 層は TopK 非適用）を導入した影響を分析する．各ニューロンが「意味的に整合的なトークン集合」にどの程度選択的に応答しているかを測る指標として，意味エントロピーを用いる．これは，当該ニューロンが高活性を示す語彙部分集合に対し，トークン埋め込み間の類似度分布のシャノンエントロピーを計算するものであり，値が小さいほど単一概念への特化を示唆する．厳密な定義と計算手順は付録 § B に示す．

図 2 は，意味エントロピーの層別分布を示す．ベースライン LM は層間変動が小さいまま，相対的に高い値（4.3–4.4）を維持するのに対し，TopK LM では，意味エントロピーが低い値（3.0–4.0）を記録しており，TopK LM のニューロンの単一概念への特化が示唆される．また，最終 2 層（非 TopK）では意味エントロピーが急上昇し，情報の再統合および抽象化が進むことと考えられる．意味エントロピーの標準偏差は特に TopK モデルでは分布の下側の裾が低く，一部のニューロンが，意味的に整合的な狭いトークン集合にほぼ排他的に応答するほど強く特化していることが示される．

表 2: 概念ステアリングの効果．単一ニューロンの活性を増幅することで，TopK LM の生成を，当該ニューロンに対応付けられた概念に関する文章へと誘導できる．第 1 行（Concept: **none**）は，増幅を行わず，プロンプト “Once upon a time” に対するモデルの基準（ベースライン）補完を示す．それ以降の各行は，特定ニューロンの増幅効果を示しており，ニューロンは **layer:index** の形式で同定される．例えば，ニューロン 22:894，すなわち第 22 層の活性化ベクトルにおける第 894 成分を増幅すると，モデルは仕事に関するテキストを生成する方向へ誘導される．

| Concept | Neuron idx | Text generated by TopK LM |
|---|---|---|
| none | — | **Once upon a time,** there were two brothers, the one being the oldest and the other the youngest. [...] |
| Work | 22:894 | **Once upon a time,** there was a huge power plant that was being pushed by a few people at one of the most challenging projects in the world. The engine was being driven by a huge engine which was being driven by a huge to be made to be driven by a vehicle which was being driven by the power of a huge drive. [...] |
| History | 18:44 | **Once upon a time,** people were called “primitive” by the Romans and by the Greeks. By the time they were called “primitive,” they were still pretty much just a bunch of different ways of thinking about things. [...] |
| Science | 17:218 | **Once upon a time,** the world was a very different place. Scientists were still very much trying to understand how the world was changing. But the world was changing too. One day, in the middle of the day, a young man from the town of Kiev came to the scientists and explained what was happening. [...] |

## 3.2 概念ステアリング

§ 3.1 の分析から，TopK LM における個々のニューロンが単義的に概念を表現していることが示唆された．本節では，単一ニューロンとモデル出力の間に因果的な結び付きを確立するため，因果介入研究 [28, 29] に従い，Activation Patching による概念ステアリング [3, 30, 31] を行う．

まず，§ 3.1 の結果から意味エントロピーの結果に基づき，意味エントロピーが低いニューロンを候補とし，平均活性を強く誘起するトークンを調べ，そのパターンに基づいて人手でラベル（例：“work”, “history”, “science”）を付与する [1)]．

介入は，単一ニューロンの活性に対して，全シーケンス位置で一定のオフセット $\delta$ を加算する操作として定義する．形式的には，トークン位置 $i$ における層 $\ell$ のニューロン $n$ での介入前の活性値を $h_{\ell,n}(i)$ とするとき，これを $h'_{\ell,n}(i) = h_{\ell,n}(i) + \delta$ で置き換える．経験的には，$\delta \in [5, 30]$ の範囲が最も顕著なステアリング効果をもたらすことが分かった．

生成時には，temperature = 0.7，top-$p$ = 0.9，top-$k$ = 50 でサンプリングし，最大 128 トークンを生成する．概念ステアリングの効果を観察するため，[29] の実験設定に従い，物語風のプロンプト（“Once upon a time,”）を入力として用いる．表 2 に示すとおり，「work」という概念に関連付けられたニューロンをステアリングすると，モデルは仕事に関連する記述を著しく多く生成するようになる．この結果は，当該ニューロンが生成に対して因果的影響を与えることを示している．

## 4 結論

本研究では, TopK 活性化機構を通じて Transformer LM に疎性を直接組み込む新たなアーキテクチャである TopK LM を提案した．事後学習を必要とする SAE とは異なり，提案手法は解釈可能性をモデルの内在的性質としつつ，SAE が抱える主要な制約を緩和する．実験の結果，TopK LM はベースライン LM と同等に競争力のある性能を達成した．さらに，ニューロンが単一概念に高く特化する傾向を示すとともに，概念ステアリングの有効性を通じて概念と出力の因果的関係を確認できることから，解釈可能性の観点でも顕著な利点を有することを示した．

解釈可能性をモデルに内在化した TopK LM は，信頼できる AI システムに向けた重要な前進である．本研究の知見は，性能と解釈可能性が必ずしも競合する目標ではないことを示しており，透明性と可制御性を備えた言語モデルに関する今後の研究に対して有望な方向性を与える．

1) ラベルの自動付与 [32] は今後の課題である．

# 謝辞

本研究は，JST/BOOST JPMJBY24F9，JST/CREST JPMJCR20D2，AMED JP25wm0625405，および，国家戦略分野の若手研究者及び博士後期課程学生の育成事業（博士後期課程学生支援）JPMJBS2421 の助成を受けたものである．また，産総研及び AIST Solutions が提供する ABCI 3.0 を利用した．

# 参考文献

[1] Lee Sharkey, Dan Braun, and Beren Millidge. Taking features out of superposition with sparse autoencoders, 2023.
[2] Hoagy Cunningham, Aidan Ewart, Logan Riggs, Robert Huben, and Lee Sharkey. Sparse Autoencoders Find Highly Interpretable Features in Language Models. In **The 12th International Conference on Learning Representations (ICLR)**, 2024.
[3] Adly Templeton, Tom Conerly, Jonathan Marcus, Jack Lindsey, Trenton Bricken, Brian Chen, Adam Pearce, Craig Citro, Emmanuel Ameisen, and et al. Scaling Monosemanticity: Extracting Interpretable Features from Claude 3 Sonnet. **arXiv preprint**, Vol. arXiv:2401.00001, , 2024.
[4] Leo Gao, Tom Dupré la Tour, Henk Tillman, Gabriel Goh, Rajan Troll, Alec Radford, Ilya Sutskever, Jan Leike, and Jeffrey Wu. Scaling and evaluating sparse autoencoders. In **The International Conference on Learning Representations (ICLR)**, 2025.
[5] Tom Lieberum, Senthooran Rajamanoharan, Arthur Conmy, Lewis Smith, Nicolas Sonnerat, Vikrant Varma, János Kramár, Anca D. Dragan, Rohin Shah, and Neel Nanda. Gemma Scope: Open Sparse Autoencoders Everywhere All At Once on Gemma 2. **arXiv preprint**, Vol. arXiv:2408.05147, , 2024.
[6] Zhengfu He, Wentao Shu, Xuyang Ge, Lingjie Chen, Junxuan Wang, Yunhua Zhou, Frances Liu, Qipeng Guo, Xuanjing Huang, and et al. Llama Scope: Extracting Millions of Features from Llama-3.1-8B with Sparse Autoencoders. **arXiv preprint**, Vol. arXiv:2410.20526, , 2024.
[7] Adam Karvonen, Can Rager, Johnny Lin, Curt Tigges, Joseph Isaac Bloom, David Chanin, Yeu-Tong Lau, Eoin Farrell, Callum McDougall, and et al. SAEBench: A Comprehensive Benchmark for Sparse Autoencoders in Language Model Interpretability. In **The 42nd International Conference on Machine Learning (ICML)**, 2025.
[8] Dong Shu, Xuansheng Wu, Haiyan Zhao, Daking Rai, Ziyu Yao, Ninghao Liu, and Mengnan Du. A Survey on Sparse Autoencoders: Interpreting the Internal Mechanisms of Large Language Models. **arXiv preprint**, Vol. arXiv:2503.05613, , 2025.
[9] Senthooran Rajamanoharan, Arthur Conmy, Lewis Smith, Tom Lieberum, Vikrant Varma, János Kramár, Rohin Shah, and Neel Nanda. Improving Dictionary Learning with Gated Sparse Autoencoders. **arXiv preprint**, Vol. arXiv:2404.16014, , 2024.
[10] Patrick Leask, Bart Bussmann, Michael T. Pearce, Joseph Isaac Bloom, Curt Tigges, Noura Al Moubayed, Lee Sharkey, and Neel Nanda. Sparse Autoencoders Do Not Find Canonical Units of Analysis. In **The International Conference on Learning Representations (ICLR)**, 2025.
[11] Sai Sumedh R. Hindupur, Ekdeep Singh Lubana, Thomas Fel, and Demba E. Ba. Projecting Assumptions: The Duality Between Sparse Autoencoders and Concept Geometry. **arXiv preprint**, Vol. arXiv:2503.01822, , 2025.
[12] Gonçalo Paulo and Nora Belrose. Sparse Autoencoders Trained on the Same Data Learn Different Features. **arXiv preprint**, Vol. arXiv:2501.16615, , 2025.
[13] Yang Xu, Yi Wang, and Hao Wang. Tracking the Feature Dynamics in LLM Training: A Mechanistic Study. **arXiv preprint**, Vol. arXiv:2412.17626, , 2024.
[14] Tatsuro Inaba, Kentaro Inui, Yusuke Miyao, Yohei Oseki, Benjamin Heinzerling, and Yu Takagi. How LLMs Learn: Tracing Internal Representations with Sparse Autoencoders. In **Proceedings of the 2025 Conference on Empirical Methods in Natural Language Processing (EMNLP)**, pp. 13458–13470, 2025.
[15] Aditya Kusupati, Vivek Ramanujan, Raghav Somani, Mitchell Wortsman, Prateek Jain, Sham Kakade, and Ali Farhadi. Soft Threshold Weight Reparameterization for Learnable Sparsity. In **The 37th International Conference on Machine Learning (ICML)**, pp. 5544–5555, 2020.
[16] Laura Graesser, Utku Evci, Erich Elsen, and Pablo Samuel Castro. The State of Sparse Training in Deep Reinforcement Learning. In **The 39th International Conference on Machine Learning (ICML)**, pp. 7766–7792, 2022.
[17] Hugo Touvron, Thibaut Lavril, Gautier Izacard, Xavier Martinet, Marie-Anne Lachaux, Timothée Lacroix, Baptiste Rozière, Naman Goyal, Eric Hambro, and et al. LLaMA: Open and Efficient Foundation Language Models. **arXiv preprint**, Vol. arXiv:2302.13971, , 2023.
[18] Guilherme Penedo, Hynek Kydlícek, Loubna Ben Allal, Anton Lozhkov, Margaret Mitchell, Colin A. Raffel, Leandro von Werra, and Thomas Wolf. The FineWeb Datasets: Decanting the Web for the Finest Text Data at Scale. In **The Thirty-Eighth Annual Conference on Neural Information Processing Systems (NeurIPS)**, 2024.
[19] Anton Lozhkov, Loubna Ben Allal, Leandro von Werra, and Thomas Wolf. FineWeb-Edu: the Finest Collection of Educational Content, 2024.
[20] Kazuki Yano, Takumi Ito, and Jun Suzuki. STEP: Staged parameter-efficient pre-training for large language models. In **Proceedings of the 2025 Conference of the Nations of the Americas Chapter of the Association for Computational Linguistics: Human Language Technologies (NAACL)**, pp. 374–384, 2025.
[21] Wataru Ikeda, Kazuki Yano, Ryosuke Takahashi, Jaesung Lee, Keigo Shibata, and Jun Suzuki. Layerwise Importance Analysis of Feed-Forward Networks in Transformer-based Language Models. 2025.
[22] Leo Gao, Jonathan Tow, Baber Abbasi, Stella Biderman, Sid Black, Anthony DiPofi, Charles Foster, Laurence Golding, Jeffrey Hsu, and et al. The Language Model Evaluation Harness, 2024.
[23] Denis Paperno, Germán Kruszewski, Angeliki Lazaridou, Quan Ngoc Pham, Raffaella Bernardi, Sandro Pezzelle, Marco Baroni, Gemma Boleda, and Raquel Fernández. The LAMBADA dataset: Word prediction requiring a broad discourse context. In **Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (ACL)**, pp. 1525–1534, 2016.
[24] Keisuke Sakaguchi, Ronan Le Bras, Chandra Bhagavatula, and Yejin Choi. WinoGrande: an adversarial winograd schema challenge at scale. **Commun. ACM**, Vol. 64, No. 9, p. 99–106, 2021.
[25] Rowan Zellers, Ari Holtzman, Yonatan Bisk, Ali Farhadi, and Yejin Choi. HellaSwag: Can a machine really finish your sentence? In **Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics (ACL)**, pp. 4791–4800, 2019.
[26] Peter Clark, Isaac Cowhey, Oren Etzioni, Tushar Khot, Ashish Sabharwal, Carissa Schoenick, and Oyvind Tafjord. Think you have solved Question Answering? Try ARC, the AI2 Reasoning Challenge. **arXiv preprint**, Vol. arXiv:1803.05457, , 2018.
[27] Todor Mihaylov, Peter Clark, Tushar Khot, and Ashish Sabharwal. Can a Suit of Armor Conduct Electricity? A New Dataset for Open Book Question Answering. In **Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing (EMNLP)**, pp. 2381–2391, 2018.
[28] Atticus Geiger, Hanson Lu, Thomas Icard, and Christopher Potts. Causal Abstractions of Neural Networks. In **Advances in Neural Information Processing Systems (NeurIPS)**, pp. 9574–9586, 2021.
[29] Alex Tamkin, Mohammad Taufeeque, and Noah D Goodman. Codebook Features: Sparse and Discrete Interpretability for Neural Networks. In **The 41st International Conference on Machine Learning (ICML)**, 2024.
[30] Arthur Conmy and Neel Nanda. Activation Steering with SAEs, 2024.
[31] Dahye Kim and Deepti Ghadiyaram. Concept Steerers: Leveraging K-Sparse Autoencoders for Controllable Generations. **arXiv preprint**, Vol. arXiv:2501.19066, , 2025.
[32] Gonçalo Paulo, Alex Mallen, Caden Juang, and Nora Belrose. Automatically Interpreting Millions of Features in Large Language Models. In **The 42nd International Conference on Machine Learning (ICML)**, 2025.
[33] Mirko Farina, Usman Ahmad, Ahmad Taha, Hussein Younes, Yusuf Mesbah, Xiao Yu, and Witold Pedrycz. Sparsity in transformers: A systematic literature review. **Neurocomputing**, Vol. 582, p. 127468, 2024.
[34] Christos Louizos, Max Welling, and Diederik P Kingma. Learning Sparse Neural Networks through L_0 Regularization. In **The 6th International Conference on Learning Representations (ICLR)**, 2018.
[35] Decebal Constantin Mocanu, Elena Mocanu, Peter Stone, Phuong H. Nguyen, Madeleine Gibescu, and Antonio Liotta. Scalable training of artificial neural networks with adaptive sparse connectivity inspired by network science. **Nature Communications**, Vol. 9, No. 2383, 2018.
[36] Utku Evci, Trevor Gale, Jacob Menick, Pablo Samuel Castro, and Erich Elsen. Rigging the Lottery: Making All Tickets Winners. In **The 37th International Conference on Machine Learning (ICML)**, pp. 2943–2952, 2020.
[37] Siddhant M. Jayakumar, Razvan Pascanu, Jack W. Rae, Simon Osindero, and Erich Elsen. Top-KAST: Top-K Always Sparse Training. In **The Thirty-Fourth Annual Conference on Neural Information Processing Systems (NeurIPS)**, 2020.
[38] Kevin Lee Hunter, Lawrence Spracklen, and Subutai Ahmad. Two sparsities are better than one: unlocking the performance benefits of sparse-sparse networks. In **Neuromorphic Computing and Engineering**, 2022.
[39] Rewon Child, Scott Gray, Alec Radford, and Ilya Sutskever. Generating Long Sequences with Sparse Transformers. **arXiv preprint**, Vol. arXiv:1904.10509, , 2019.
[40] Ankit Gupta, Guy Dar, Shaya Goodman, David Ciprut, and Jonathan Berant. Memory-efficient Transformers via Top-k Attention. In **SustaiNLP@EMNLP**, 2021.
[41] Noam Shazeer, Azalia Mirhoseini, Krzysztof Maziarz, Andy Davis, Quoc V. Le, Geoffrey E. Hinton, and Jeff Dean. Outrageously Large Neural Networks: The Sparsely-Gated Mixture-of-Experts Layer. In **The 5th International Conference on Learning Representations (ICLR)**, 2017.
[42] Sebastian Jaszczur, Aakanksha Chowdhery, Afroz Mohiuddin, Łukasz Kaiser, Wojciech Gajewski, Henryk Michalewski, and Jonni Kanerva. Sparse is Enough in Scaling Transformers. In **The Thirty-Fifth Annual Conference on Neural Information Processing Systems (NeurIPS)**, 2021.
[43] Alireza Makhzani and Brendan J Frey. Winner-Take-All Autoencoders. In **Advances in Neural Information Processing Systems (NIPS)**, 2015.
[44] Subutai Ahmad and Luiz Scheinkman. How Can We Be So Dense? The Benefits of Using Highly Sparse Representations. **arXiv preprint**, Vol. arXiv:1903.11257, , 2019.
[45] Lorenz Kuhn, Yarin Gal, and Sebastian Farquhar. Semantic Uncertainty: Linguistic Invariances for Uncertainty Estimation in Natural Language Generation. In **The Eleventh International Conference on Learning Representations (ICLR)**, 2023.
[46] Sebastian Farquhar, Jannik Kossen, Lorenz Kuhn, and Yarin Gal. Detecting hallucinations in large language models using semantic entropy. **Nature**, Vol. 630, No. 8017, pp. 625–630, 2024.

## A　関連研究

ニューラルネットワークの文脈において，**疎性（sparsity）** という語は相互に関連しつつも異なる複数の意味で用いられる [33]．具体的には，モデル重みの疎性 [34–38]，疎な注意機構 [39, 40]，Mixture-of-Experts（MoE）型アーキテクチャにおける構成要素の疎な活性化 [41, 42]，そして SAE で典型的に扱われる，隠れ状態活性の表現的疎性 [1–4, 38, 43, 44] が挙げられる．本研究はこのうち最後のカテゴリ，すなわち表現的疎性に属する．

表現的疎性に関する関連研究のうち，本研究に最も近いものとして codebook feature layers [29] がある．同手法は，本研究の TopK 層と同様に，本質的な解釈可能性を備える点で共通する．一方で，疎性の実現方法は異なる．codebook feature layers はベクトル量子化および最大内積探索に依拠するのに対し，本研究で提案するアーキテクチャは Transformer への最小限の改変のみで実現できる．

## B　意味エントロピー

ニューロン活性が語彙のうち意味的に整合的なクラスタ上でどの程度「広がっているか」を定量化するため，**意味エントロピー（Semantic Entropy）** を定義する．先行研究では文レベルの意味的不確実性の尺度が提案されているが [45, 46]，本研究ではこの考え方をニューロンレベルへ適用し，各隠れニューロンが選択的に応答する「意味に関係したトークン集合」を定量評価可能にする．

層インデックスを $\ell$，ニューロンインデックスを $n$ とする．まず各ニューロン $(\ell, n)$ について，平均活性が閾値を上回るトークン集合を同定する．ニューロン $(\ell, n)$ のトークン $t$ に対する平均活性を $A_{n,t}^{(\ell)}$ とし，$\theta$ を $A_{n,t}^{(\ell)}$ 全体の 99.9 パーセンタイル値の 70% と定める．このとき，選択される語彙部分集合を次式で定義する：

$$V_{\ell,n} = \{\, t \mid A_{n,t}^{(\ell)} > \theta \}. \tag{4}$$

次に，$V_{\ell,n}$ に含まれるトークン間の意味的一貫性を捉えるため，各トークンの埋め込みベクトル $e_t$ を取り出し，トークン対のコサイン類似度を計算する：

$$S_{t,t'} = \cos(e_t, e_{t'}) \quad \forall t, t' \in V_{\ell,n} \tag{5}$$

より頻出するトークンが意味構造に比例的に寄与すべきであるため，トークン対 $(t, t')$ に対し，それぞれのコーパス頻度 $f_t$ および $f_{t'}$ の積で重み付けを行う：

$$w_{t,t'} = f_t \cdot f_{t'} \quad \forall t, t' \in V_{\ell,k} \tag{6}$$

続いて，コサイン類似度の値域を $n_{\text{bins}}$[2] 個の連続ビン $\{B_i\}$ に分割し，ビン上の重み付きヒストグラム分布を構成する：

$$p_i = \sum_{(t,t') \in B_i} \frac{w_{t,t'}}{\sum_{t,t' \in V_{\ell,k}} w_{t,t'}} \tag{7}$$

最後に，ニューロン $(\ell, n)$ の意味エントロピーを，このビン重み付き類似度分布のシャノンエントロピーとして定義する：

$$H_{\text{sem}}(\ell, k) = -\sum_{i=1}^{n_{\text{bins}}} p_i \log_2 p_i \tag{8}$$

$H_{\text{sem}}(\ell, k)$ が小さいほど，高活性が意味的に整合的なトークン部分集合に集中しており，当該ニューロンが意味的に選択的であることを示す．逆に，値が大きい（最大で $\log_2 n_{\text{bins}}$）場合，活性が複数の意味クラスタにまたがって広がっていることを意味し，当該ニューロンがより汎用的な応答を示すことを示唆する．

2）　実験では $n_{\text{bins}} = 1000$ とした．