Transformer LLM の内部挙動改善：隠れ状態ベクトルの数値的収束性の向上

柴田 圭悟 [1]　高橋 良允 [1]　矢野 一樹 [1]　李 宰成 [1]　池田 航 [1]　鈴木 潤 [1,2,3]
[1] 東北大学　[2] 理化学研究所　[3] 国立情報学研究所
shibata.keigo.p1@dc.tohoku.ac.jp

概要

Transformer の事前学習済みモデルでは，各層の予測分布の意味的収束性が示される一方で，隠れ状態ベクトルの数値的収束性が確認されておらず，両者に乖離があることが問題である．本研究では，事前学習済みモデルに自己蒸留を適用し，隠れ状態ベクトルの数値的収束性を改善する．自己蒸留は，同一モデル内で深い層から浅い層へ知識を蒸留し，モデルの性能を維持しながら隠れ状態ベクトルの収束性を高めることが可能である．提案手法により，最終層付近での隠れ状態ベクトルの収束性が向上した．

1　はじめに

Transformer の隠れ状態ベクトルから，意思決定過程の詳細な解析や解釈が可能である．特に，各層の隠れ状態ベクトルを語彙次元に射影することで，Transformer 内部で次単語予測の推論過程を可視化できる [1, 2]．この手法により，各層で予測される単語が最終層で予測される単語に意味的に近づいていく傾向が観測できる．

これらの研究から各層の予測分布の意味的な収束性は示唆されるが，一方で，隠れ状態ベクトル自体の数値的な収束性は確認されていない．実際，コサイン類似度を用いた調査により，隠れ状態は中間層ではほとんど変化せず，最終層付近で大きく変化することが示されている [3]．この結果は，中間層のモデルへの寄与が小さい一方で，最終層付近では隠れ状態が大きく変化していることを示唆しており，予測分布に基づく結果とは異なる性質を持つ．直感的には，モデルの浅い層では大まかな次単語予測を行うために隠れ状態が大きく変化し，深い層では予測を微調整する段階に移行するため，隠れ状態の変化は小さくなることが期待される（図 1）．この隠れ状態ベクトルの数値的な遷移を「隠れ状態ベクトルの収束性」と定義する．

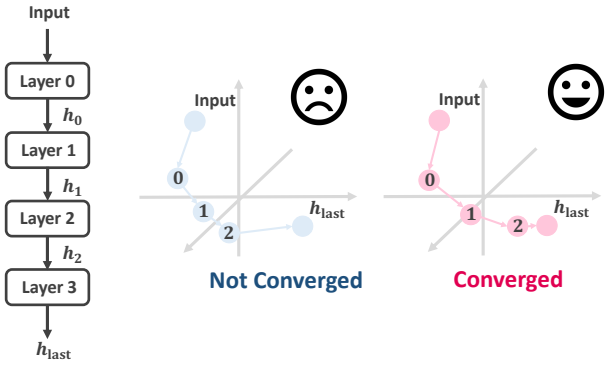


図 1: Transformer の隠れ状態ベクトルの遷移．事前学習済みの隠れ状態ベクトル（青点）で，中間層が隠れ状態の変化に寄与せずに，最終層で大きな変化が起こっている．一方で，自己蒸留後のモデルの隠れ状態ベクトル（赤点）は，深い層ほど隠れ状態の変化が小さくなっており，収束性が改善している．

本研究の目標は，予測分布の意味的収束性と隠れ状態ベクトルの数値的な収束性の乖離に対処し，隠れ状態ベクトルの数値的収束性を改善することで，より解釈性の高いモデルを構築することである．意味的収束性を保ちながら，隠れ状態ベクトルの数値的な収束性を改善するために，中間表現を利用して，同一モデル内で知識蒸留を行う自己蒸留 [4] を事前学習済みモデルに対して行う．この結果，最終層付近での隠れ状態ベクトルの移動量が減少し，隠れ状態ベクトルの収束性が改善された（図 3）．さらに，意味的収束性との整合性が確認された（図 7）．

2　隠れ状態ベクトルの収束性

Transformer による LLM が文章を生成する処理は，次のトークンを予測する計算（next token prediction）の自己回帰的な処理で成り立っている．本研究では，次のトークンを予測する処理を，入力層の入力トークンの埋め込みベクトルを出発点として，出力

層の次トークンの埋め込みベクトル（の近傍）を到達点とした各層の隠れベクトルによる埋め込み空間内の移動と捉える．この時，埋め込みベクトル空間内の移動軌跡（内部挙動）に関して，ランダムに移動するよりも収束性の性質があることが望ましい．埋め込みベクトル空間内において，各層の隠れ状態ベクトルの移動軌跡に関する収束性を以下のように定義する．

定義 1 (隠れ状態ベクトルの移動量)**.** モデルの層数を L とする．層 l（$0 \le l \le L$）の入出力を h_{l-1}, h_l とし（層 0 の入力は Input とする），層 l での隠れ状態ベクトルの移動量を次のように定義する．

$$D_l = 1 - \frac{1 + \text{cosine\_similarity}(h_{l-1}, h_l)}{2} \tag{1}$$

ここで，cosine_similarity は 2 ベクトルのコサイン類似度を求める関数であり，D_l は $0 \le D_l \le 1$ の範囲に収まる．

定義 2 (隠れ状態ベクトルの収束性)**.** 層番号 l（$0 \le l \le L$）における移動量 D_l の単調減少性を評価する指標 S_{model} を式 2 に定義する．D_l が単調減少に近いほど，S_{model} の値は 1 に近づく．

$$S_{\text{model}} = 1 - \frac{\sum_{l=1}^{L} \max(0, D_l - D_{l-1})}{\sum_{l=1}^{L} |D_l - D_{l-1}|} \tag{2}$$

3 自己蒸留（Self Distillation）

自己蒸留は，深い層を教師モデル，浅い層を生徒モデルとして，深い層から浅い層へ知識転移を手法である．[4]．別々の教師モデル，生徒モデルを用意して教師モデルから生徒モデルへ知識転移する知識蒸留に比べて，低コストで高い性能を達成できる．

本研究で用いる，モデルの自己蒸留方法を図 2 に示す．損失関数は，最終層の予測分布と，各層の予測分布の KL ダイバージェンスの重み付き和（式 4）と，隣接した隠れ状態の二乗ノルムの重み付き和（式 5）である．重み w_l は，総和が 1，深い層ほど大きくなるように設定した（式 6）．式 4 は，各層の語彙分布が教師モデルである最終層の語彙分布に似るようにするための項である．本来の自己蒸留の文脈では，最終層の語彙分布 q_L との KL ダイバージェンスを設定するが，事前学習済みモデルの出力 q_{target} との KL ダイバージェンスとすることで，性能劣化を防ぐ．式 5 は，隣接した隠れ状態ベクトルの距離を近づけることで，隠れ状態ベクトルを収束させることを意図している．

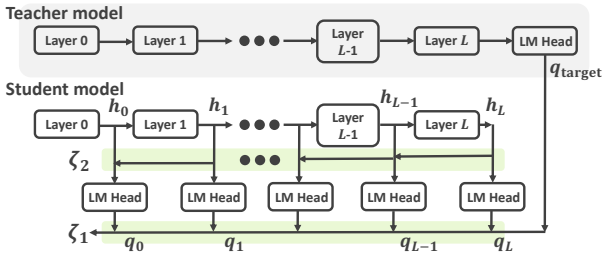


図 2: 自己蒸留の損失関数

$$\zeta = \zeta_1 + \alpha\zeta_2 \tag{3}$$

$$\zeta_1 = \sum_{l=0}^{L} w_l \text{KL}(q_l, q_{\text{target}}) \tag{4}$$

$$\zeta_2 = \sum_{l=1}^{L} w_l \frac{||h_l - h_{l-1}||_2}{||h_{l-1}||_2} \tag{5}$$

$$w_l = \frac{\exp(kl)}{\sum_{i=0}^{L} \exp(ki)} \tag{6}$$

ただし，k は重み係数であり，人手により決定するハイパーパラメータである．

4 実験

実験では，通常の事前学習済みモデルと，そのモデルに対して追加で自己蒸留を行ったモデルの隠れ状態ベクトルの収束性の違いを調べ，自己蒸留をモデルに対して追加で行うことで，収束性が改善することを示す．

4.1 実験設定

モデル． Llama3[5] の 1B, 3B，Qwen2[6] の 0.5B, 1.5B，3B を使用した．Qwen2 の実験結果は付録 A に記載した．

評価尺度． 隠れ状態ベクトルの収束性は，移動量（定義 1）D_l の単調減少性（定義 2）で評価する．この時，隠れ状態ベクトルを主成分分析し，収束性の定性的な分析も合わせて実施する．

予測分布の意味的な収束性は，中間層の隠れ状態ベクトルを LM ヘッドに通した語彙空間の分布から，各層のタスクの予測性能と，推論過程の可視化より評価する．

評価用データセット． モデルの性能評価には 2 つのデータセットを使用する．長文の最後の単語を予測し広範囲の文脈理解力を測る **LAMBADA**[7] から次単語予測の精度，wikipedia から構築された **Wikitext**[8] から Perplexity（PPL）を測る．評価に

は，lm-evaluation-harness ライブラリ [9] を使用した．

4.2 実験手順

ベースラインの Llama-3.2-1B, Llama-3.2-3B に対して，Fineweb-Edu [10] から抽出した 10M トークンを式 3 の損失関数で自己蒸留を行ったモデルを SelfDistillationLlama とする．バッチサイズ 128, 学習率は 1e-4，コサインスケジューラ，式 3 の係数 α は 1, 式 6 の係数 k は 0.25 とした．

4.3 実験結果

4.3.1 隠れ状態ベクトルの収束性の評価

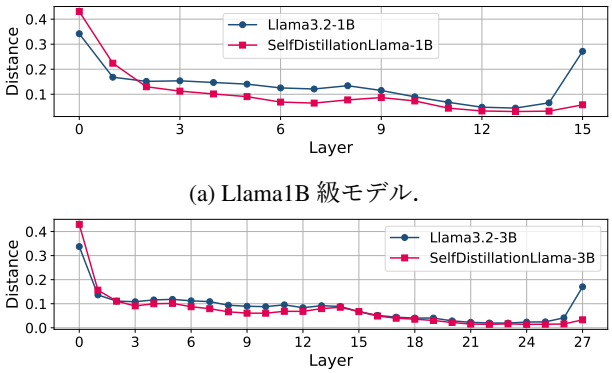
Llama3.2-1B と Llama3.2-3B と，それぞれを自己蒸留した際の各層の隠れ状態ベクトルの移動量（定義 1）を図 3 に示す．収束性（定義 2）の指標では，1B 級のモデルでは $S_{\text{Llama3.2-1B}} = 0.56$, $S_{\text{SelfLlama-1B}} = 0.90$ に，3B 級のモデルでは $S_{\text{Llama3.2-3B}} = 0.66$, $S_{\text{SelfLlama-3B}} = 0.89$ へと改善がみられた．特に，図 3 から，最終層での移動量が大幅に改善されたことがわかる．1B 級のモデルでは，ベースラインの最終層での移動量は 0.27 であり，前の層から約 4.14 倍に増加している．一方，SelfDistillationLlama では，最終層での移動量が 0.058 まで減少し，前の層からの増加は 1.69 倍に留まる．3B 級のモデルでも同様の改善が確認できる．この結果は，自己蒸留により最終層での過剰な変化が抑制され，隠れ状態ベクトルの収束性が向上したことを示している．

Llama3.2-1B との SelfDistillationLlama-1B の各層の隠れ状態ベクトルを主成分分析によって 3 次元に可視化した結果を図 4 に示す．Llama3.2-1B は最終層で状態が大きく変化する様子が確認できる（図 4a）．一方，SelfDistillationLlama-1B では，最終層付近で隠れ状態が収束していることが示されており（図 4b），定性的にも隠れ状態ベクトルの収束性が改善されたことがわかる．

4.3.2 意味的な収束性の評価

図 5 に，lambada データセットによる各層の次単語予測性能の評価，図 6 に，wikitext データセットによる Perplexity（PPL）の評価を示す．

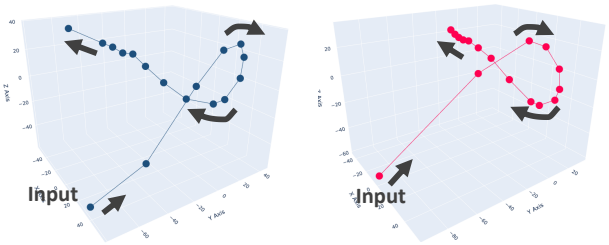
1B 級，3B 級のモデルは，ベースラインよりも最終層付近で性能向上が確認された．具体的には，1B 級のモデルにおいて，Llama3.2-1B は次単語予測の性能が 15 層（最終層）と 14 層で約 38%減少している．一方，SelfDistillationLlama-1B は，次単語予測の精度が 15 層と 14 層での減少率が 9%に抑えられており，最終層を除く全ての層でベースラインより高い精度を維持している．（図 5a）最終層の精度がベースラインを下回った要因としては，少数のトークンで事前学習済みモデルを自己蒸留してモデルを構築したことが影響していると考えられる．また，wikitext の PPL は，ベースラインよりも一貫して小さな値を示している．



(a) Llama1B 級モデル．

(b) Llama3B 級モデル．

図 3: 隠れ状態ベクトルの移動量の遷移．



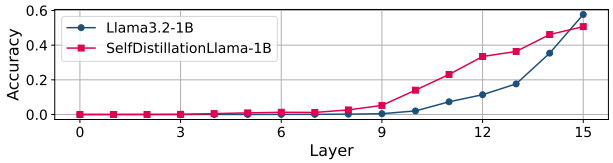
(a) Llama3.2-1B　(b) SelfDistillationLlama-1B

図 4: Llama1B 級モデルの各層の隠れ状態の可視化．

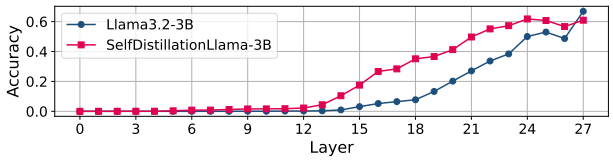
図 7 に，Llama1B 級のモデルに文章を与えた際の各層の推論過程の可視化および隠れ状態ベクトルの移動量を示す．Llama3.2-1B では，層が深くなるにつれて単語の意味的な収束性が確認できる一方で，最終層において隠れ状態ベクトルが大きく変化しており，意味的収束性と隠れ状態ベクトルの数値的収束性の間に乖離が見られる．これに対し，自己蒸留を適用したモデルでは，隠れ状態の数値的な収束性も確認できる．

5 関連研究

Transformer の反復推論． Transformer の最終層の表現は，浅い層から深い層にかけての隠れ状態から段階的に構築されている [1, 2, 11]．これは，ResNets

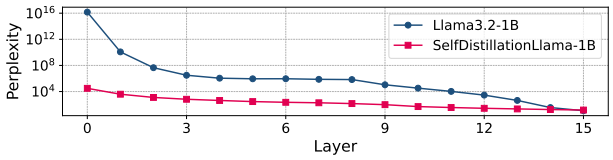


(a) Llama1B 級モデル.

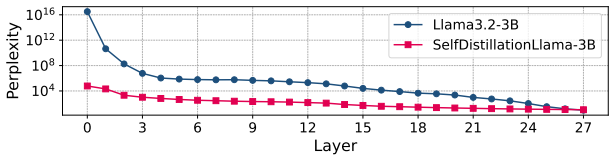


(b) Llama3B 級モデル.

図 5: Lambada データセットによる次単語予測の性能.



(a) Llama1B 級モデル.



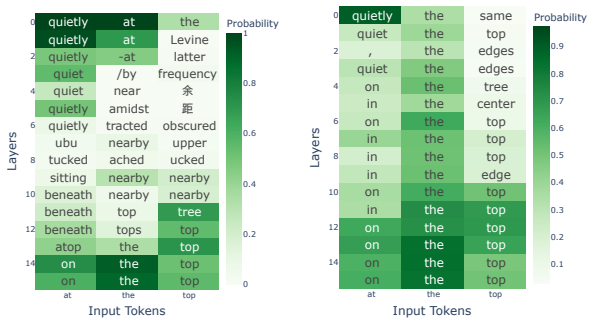
(b) Llama3B 級モデル.

図 6: wikitext データセットによる Perplexity.

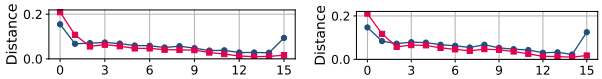
の残差結合が，自然に隠れ状態の反復的な更新を促す [12, 13] という結果が Transformer でも適用可能であることを示している．この反復推論は，層を削除すると，後続の層が挙動を変化させて自己修復する観点からも支持されている [14].

モデルの冗長性． Transformer の中間層が，入力に近い初期層や出力に近い最終層付近の層に比べて性能に影響しないことを，層の削除や入れ替えによって実験的に示している [15, 16, 3]．また，Pruning によって，BERT の約 85%のニューロンが冗長であること [17] や，66B 級のモデルで性能の劣化を最小に抑えながら，Attention head の 70%, Feed Forward 層の 20%のパラメータを削除できること [18] が報告されている．Transformer に冗長性が生まれることを抑制するために，事前学習時に層をランダムで削除する LayerDrop [19, 20] や，層をシャッフルする LayerShuffle [21] などの手法が提案されている．

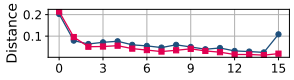
早期終了（early exit）． 中間層の隠れ状態から最終層の表現を直接予測することで早期終了を可能にする手法が提案されており，推論の高速化が可能である．学習済みモデルを中間層の隠れ状態が最終層の表現に近づくように継続学習させる手法 [22, 23] や，学習済みモデルの外部に学習可能な線形層などを通じて中間層の隠れ状態から最終層の表現を予測する手法 [24, 25] が提案されている．



(a) Llama3.2-1B　(b) SelfDistillationLlama-1B



(c) at　(d) the



(e) top

図 7: Llama1B のモデルに「The curious cat climbed up the tall tree, carefully balancing on each thin branch, until it suddenly noticed a small bird sitting quietly at the top.」の最後の 3 トークンである「at the top」を予測させた際の推論過程の可視化と各トークンを予測する際の隠れ状態の移動量.

6　おわりに

本研究では，Llama や Qwen といった事前学習済みモデルにおいて，隠れ状態ベクトルが最終層で大きく変化し，数値的収束性を持たないという課題に着目した．この課題を解決するため，事前学習済みモデルに対して，深い層から浅い層へ知識を蒸留する自己蒸留を適用し，隠れ状態ベクトルの数値的収束性が改善されることを示した．今後の課題として，収束性を高めた後の層削除によるモデルの軽量化を検討する．また，隠れ状態ベクトルが収束性を持つような新しい事前学習手法の開発にも取り組みたい．本研究で得られた知見が，より質の高い LLM 開発技術への一助となることを期待する．

謝辞

本研究は，JST ムーンショット型研究開発事業 JPMJMS2011-35 (fundamental research)，および，文部科学省の補助事業「生成 AI モデルの透明性・信頼性の確保に向けた研究開発拠点形成」の支援を受けたものです．また，本研究は九州大学情報基盤研究開発センター研究用計算機システムの一般利用を利用したものです．本研究成果の一部は，データ活用社会創成プラットフォーム mdx[26] を利用して得られたものです．

参考文献

[1] Nostalgebraist. Interpreting gpt: The logit lens. https://www.alignmentforum.org/posts/AcKRB8wDpdaN6v6ru/interpreting-gpt-the-logit-lens, 2020.

[2] Nora Belrose, Zach Furman, Logan Smith, Danny Halawi, Igor Ostrovsky, Lev McKinney, Stella Biderman, and Jacob Steinhardt. Eliciting latent predictions from transformers with the tuned lens. **CoRR**, Vol. abs/2303.08112, , 2023.

[3] Qi Sun, Marc Pickett, Aakash Kumar Nain, and Llion Jones. Transformer layers as painters. **CoRR**, Vol. abs/2407.09298, , 2024.

[4] Linfeng Zhang, Jiebo Song, Anni Gao, Jingwei Chen, Chenglong Bao, and Kaisheng Ma. Be your own teacher: Improve the performance of convolutional neural networks via self distillation. In **2019 IEEE/CVF International Conference on Computer Vision, ICCV 2019, Seoul, Korea (South), October 27 - November 2, 2019**, pp. 3712–3721. IEEE, 2019.

[5] Abhimanyu Dubey, Abhinav Jauhri, Abhinav Pandey, Abhishek Kadian, Ahmad Al-Dahle, Aiesha Letman, Akhil Mathur, Alan Schelten, Amy Yang, Angela Fan, Anirudh Goyal, Anthony Hartshorn, Aobo Yang, Archi Mitra, Archie Sravankumar, Artem Korenev, Arthur Hinsvark, Arun Rao, Aston Zhang, Aurélien Rodriguez, Austen Gregerson, Ava Spataru, Baptiste Rozière, Bethany Biron, Binh Tang, Bobbie Chern, Charlotte Caucheteux, Chaya Nayak, Chloe Bi, Chris Marra, Chris McConnell, Christian Keller, Christophe Touret, Chunyang Wu, Corinne Wong, Cristian Canton Ferrer, Cyrus Nikolaidis, Damien Allonsius, Daniel Song, Danielle Pintz, Danny Livshits, David Esiobu, Dhruv Choudhary, Dhruv Mahajan, Diego Garcia-Olano, Diego Perino, Dieuwke Hupkes, Egor Lakomkin, Ehab AlBadawy, Elina Lobanova, Emily Dinan, Eric Michael Smith, Filip Radenovic, Frank Zhang, Gabriel Synnaeve, Gabrielle Lee, Georgia Lewis Anderson, Graeme Nail, Grégoire Mialon, Guan Pang, Guillem Cucurell, Hailey Nguyen, Hannah Korevaar, Hu Xu, Hugo Touvron, Iliyan Zarov, Imanol Arrieta Ibarra, Isabel M. Kloumann, Ishan Misra, Ivan Evtimov, Jade Copet, Jaewon Lee, Jan Geffert, Jana Vranes, Jason Park, Jay Mahadeokar, Jeet Shah, Jelmer van der Linde, Jennifer Billock, Jenny Hong, Jenya Lee, Jeremy Fu, Jianfeng Chi, Jianyu Huang, Jiawen Liu, Jie Wang, Jiecao Yu, Joanna Bitton, Joe Spisak, Jongsoo Park, Joseph Rocca, Joshua Johnstun, Joshua Saxe, Junteng Jia, Kalyan Vasuden Alwala, Kartikeya Upasani, Kate Plawiak, Ke Li, Kenneth Heafield, Kevin Stone, and et al. The llama 3 herd of models. **CoRR**, Vol. abs/2407.21783, , 2024.

[6] An Yang, Baosong Yang, Binyuan Hui, Bo Zheng, Bowen Yu, Chang Zhou, Chengpeng Li, Chengyuan Li, Dayiheng Liu, Fei Huang, Guanting Dong, Haoran Wei, Huan Lin, Jialong Tang, Jialin Wang, Jian Yang, Jianhong Tu, Jianwei Zhang, Jianxin Ma, Jianxin Yang, Jin Xu, Jingren Zhou, Jinze Bai, Jinzheng He, Junyang Lin, Kai Dang, Keming Lu, Keqin Chen, Kexin Yang, Mei Li, Mingfeng Xue, Na Ni, Pei Zhang, Peng Wang, Ru Peng, Rui Men, Ruize Gao, Runji Lin, Shijie Wang, Shuai Bai, Sinan Tan, Tianhang Zhu, Tianhao Li, Tianyu Liu, Wenbin Ge, Xiaodong Deng, Xiaohuan Zhou, Xingzhang Ren, Xinyu Zhang, Xipin Wei, Xuancheng Ren, Xuejing Liu, Yang Fan, Yang Yao, Yichang Zhang, Yu Wan, Yunfei Chu, Yuqiong Liu, Zeyu Cui, Zhenru Zhang, Zhifang Guo, and Zhihao Fan. Qwen2 technical report. **CoRR**, Vol. abs/2407.10671, , 2024.

[7] 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 2016, August 7-12, 2016, Berlin, Germany, Volume 1: Long Papers**. The Association for Computer Linguistics, 2016.

[8] Stephen Merity, Caiming Xiong, James Bradbury, and Richard Socher. Pointer sentinel mixture models. In **5th International Conference on Learning Representations, ICLR 2017, Toulon, France, April 24-26, 2017, Conference Track Proceedings**. OpenReview.net, 2017.

[9] Leo Gao, Jonathan Tow, Baber Abbasi, Stella Biderman, Sid Black, Anthony DiPofi, Charles Foster, Laurence Golding, Jeffrey Hsu, Alain Le Noac'h, Haonan Li, Kyle McDonell, Niklas Muennighoff, Chris Ociepa, Jason Phang, Laria Reynolds, Hailey Schoelkopf, Aviya Skowron, Lintang Sutawika, Eric Tang, Anish Thite, Ben Wang, Kevin Wang, and Andy Zou. A framework for few-shot language model evaluation, 07 2024.

[10] Anton Lozhkov, Loubna Ben Allal, Leandro von Werra, and Thomas Wolf. Fineweb-edu: the finest collection of educational content, 2024.

[11] Mor Geva, Avi Caciularu, Kevin Ro Wang, and Yoav Goldberg. Transformer feed-forward layers build predictions by promoting concepts in the vocabulary space. In Yoav Goldberg, Zornitsa Kozareva, and Yue Zhang, editors, **Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing, EMNLP 2022, Abu Dhabi, United Arab Emirates, December 7-11, 2022**, pp. 30–45. Association for Computational Linguistics, 2022.

[12] Klaus Greff, Rupesh Kumar Srivastava, and Jürgen Schmidhuber. Highway and residual networks learn unrolled iterative estimation. In **5th International Conference on Learning Representations, ICLR 2017, Toulon, France, April 24-26, 2017, Conference Track Proceedings**. OpenReview.net, 2017.

[13] Stanislaw Jastrzebski, Devansh Arpit, Nicolas Ballas, Vikas Verma, Tong Che, and Yoshua Bengio. Residual connections encourage iterative inference. In **6th International Conference on Learning Representations, ICLR 2018, Vancouver, BC, Canada, April 30 - May 3, 2018, Conference Track Proceedings**. OpenReview.net, 2018.

[14] Cody Rushing and Neel Nanda. Explorations of self-repair in language models. In **Forty-first International Conference on Machine Learning, ICML 2024, Vienna, Austria, July 21-27, 2024**. OpenReview.net, 2024.

[15] Vedang Lad, Wes Gurnee, and Max Tegmark. The remarkable robustness of llms: Stages of inference? **CoRR**, Vol. abs/2406.19384, , 2024.

[16] Xin Men, Mingyu Xu, Qingyu Zhang, Bingning Wang, Hongyu Lin, Yaojie Lu, Xianpei Han, and Weipeng Chen. Shortgpt: Layers in large language models are more redundant than you expect. **CoRR**, Vol. abs/2403.03853, , 2024.

[17] Fahim Dalvi, Hassan Sajjad, Nadir Durrani, and Yonatan Belinkov. Analyzing redundancy in pretrained transformer models. In Bonnie Webber, Trevor Cohn, Yulan He, and Yang Liu, editors, **Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing, EMNLP 2020, Online, November 16-20, 2020**, pp. 4908–4926. Association for Computational Linguistics, 2020.

[18] Hritik Bansal, Karthik Gopalakrishnan, Saket Dingliwal, Sravan Bodapati, Katrin Kirchhoff, and Dan Roth. Rethinking the role of scale for in-context learning: An interpretability-based case study at 66 billion scale. In Anna Rogers, Jordan L. Boyd-Graber, and Naoaki Okazaki, editors, **Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), ACL 2023, Toronto, Canada, July 9-14, 2023**, pp. 11833–11856. Association for Computational Linguistics, 2023.

[19] Minjia Zhang and Yuxiong He. Accelerating training of transformer-based language models with progressive layer dropping. In Hugo Larochelle, Marc'Aurelio Ranzato, Raia Hadsell, Maria-Florina Balcan, and Hsuan-Tien Lin, editors, **Advances in Neural Information Processing Systems 33: Annual Conference on Neural Information Processing Systems 2020, NeurIPS 2020, December 6-12, 2020, virtual**, 2020.

[20] Angela Fan, Edouard Grave, and Armand Joulin. Reducing transformer depth on demand with structured dropout. In **8th International Conference on Learning Representations, ICLR 2020, Addis Ababa, Ethiopia, April 26-30, 2020**. OpenReview.net, 2020.

[21] Matthias Freiberger, Peter Kun, Anders Sundnes Løvlie, and Sebastian Risi. Layershuffle: Enhancing robustness in vision transformers by randomizing layer execution order. **CoRR**, Vol. abs/2407.04513, , 2024.

[22] Mostafa Elhoushi, Akshat Shrivastava, Diana Liskovich, Basil Hosmer, Bram Wasti, Liangzhen Lai, Anas Mahmoud, Bilge Acun, Saurabh Agarwal, Ahmed Roman, Ahmed A Aly, Beidi Chen, and Carole-Jean Wu. Layerskip: Enabling early exit inference and self-speculative decoding. In Lun-Wei Ku, Andre Martins, and Vivek Srikumar, editors, **Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), ACL 2024, Bangkok, Thailand, August 11-16, 2024**, pp. 12622–12642. Association for Computational Linguistics, 2024.

[23] Fulai Nan, Jin Wang, and Xuejie Zhang. An on-device machine reading comprehension model with adaptive fast inference. In Wei Lu, Shujian Huang, Yu Hong, and Xiabing Zhou, editors, **Natural Language Processing and Chinese Computing - 11th CCF International Conference, NLPCC 2022, Guilin, China, September 24-25, 2022, Proceedings, Part I**, Vol. 13551 of **Lecture Notes in Computer Science**, pp. 850–862. Springer, 2022.

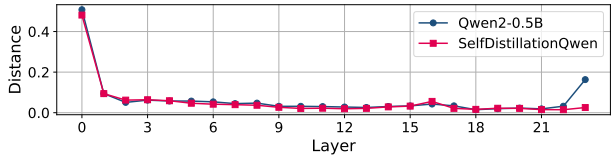
[24] Ariel Gera, Roni Friedman, Ofir Arviv, Chulaka Gunasekara, Benjamin Sznajder, Noam Slonim, and Eyal Shnarch. The benefits of bad advice: Autocontrastive decoding across model layers. In Anna Rogers, Jordan L. Boyd-Graber, and Naoaki Okazaki, editors, **Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), ACL 2023, Toronto, Canada, July 9-14, 2023**, pp. 10406–10420. Association for Computational Linguistics, 2023.

[25] Alexander Yom Din, Taelin Karidi, Leshem Choshen, and Mor Geva. Jump to conclusions: Short-cutting transformers with linear transformations. In Nicoletta Calzolari, Min-Yen Kan, Véronique Hoste, Alessandro Lenci, Sakriani Sakti, and Nianwen Xue, editors, **Proceedings of the 2024 Joint International Conference on Computational Linguistics, Language Resources and Evaluation, LREC/COLING 2024, 20-25 May, 2024, Torino, Italy**, pp. 9615–9625. ELRA and ICCL, 2024.

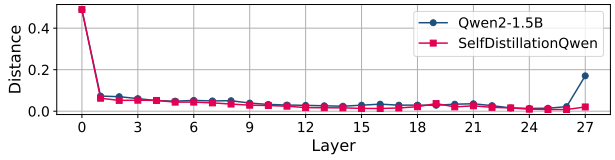
[26] Toyotaro Suzumura, Akiyoshi Sugiki, Hiroyuki Takizawa, Akira Imakura, Hiroshi Nakamura, Kenjiro Taura, Tomohiro Kudoh, Toshihiro Hanawa, Yuji Sekiya, Hiroki Kobayashi, Yohei Kuga, Ryo Nakamura, Renhe Jiang, Junya Kawase, Masatoshi Hanai, Hiroshi Miyazaki, Tsutomu Ishizaki, Daisuke Shimotoku, Daisuke Miyamoto, Kento Aida, Atsuko Takefusa, Takashi Kurimoto, Koji Sasayama, Naoya Kitagawa, Ikki Fujiwara, Yusuke Tanimura, Takayuki Aoki, Toshio Endo, Satoshi Ohshima, Keiichiro Fukazawa, Susumu Date, and Toshihiro Uchibayashi. mdx: A cloud platform for supporting data science and cross-disciplinary research collaborations. In **2022 IEEE Intl Conf on Dependable, Autonomic and Secure Computing, Intl Conf on Pervasive Intelligence and Computing, Intl Conf on Cloud and Big Data Computing, Intl Conf on Cyber Science and Technology Congress (DASC/PiCom/CBDCom/CyberSciTech)**, pp. 1–7, 2022.

A　Qwen2 の実験結果

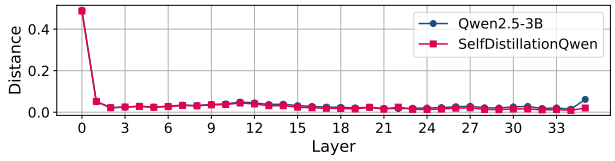
Qwen2 の 0.5B, 1.5B, 3B で Llama3 モデルと同様の実験を行った結果を図 8, 図 9, 図 10 にそれぞれ示す.



(a) Qwen2-0.5B

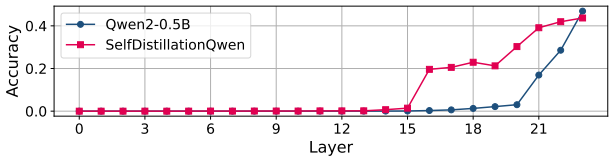


(b) Qwen2-1.5B

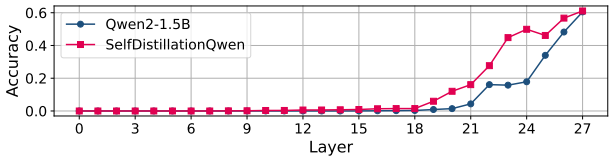


(c) Qwen2-3B

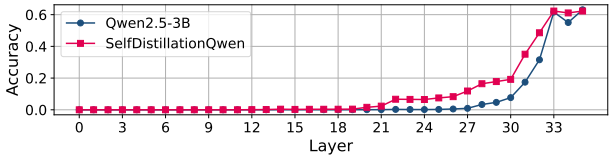
図 8: 隠れ状態ベクトルの移動量の遷移.



(a) Qwen2-0.5B

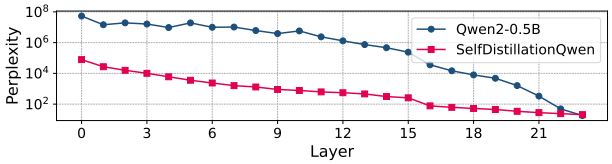


(b) Qwen2-1.5B

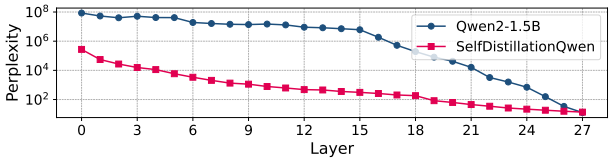


(c) Qwen2-3B

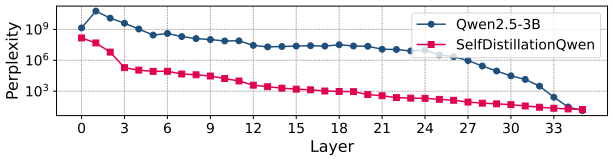
図 9: Lambada データセットによる次単語予測の性能.



(a) Qwen2-0.5B



(b) Qwen2-1.5B



(c) Qwen2-3B

図 10: wikitext データセットによる Perplexity.