ロススパイクの影響分析

杉浦一瑳 ♣,‡, 栗田修平 ◇,‡, 小田 悠介 ‡
♣ 京都大学, ◇ 国立情報学研究所,
‡ 国立情報学研究所 大規模言語モデル研究開発センター
sugiura.issa.q29@kyoto-u.jp {skurita, odashi}@nii.ac.jp

概要

ニューラルネットワークの学習中に突然損失が発散する現象はロススパイクと呼ばれ，学習が破綻する原因として知られている．モデルの学習には大きなコストがかかるため，ロススパイクの発生を予防する方法が数多く提案されてきた一方で，スパイクがモデルに与える影響については十分に理解されていない．本稿では，ロススパイクのモデルへの影響について，2 つのモデル間の損失地形における結びつきを表す線形峰接続性の観点から分析する．小規模な言語モデルをロススパイクが発生する設定を含む複数の学習設定で事前学習し，学習中のチェックポイントを用いてスパイクの影響を分析した．その結果，パラメータがスパイクの前後で大きく異なる位置に変化すること，及び線形峰接続性のパターンが変化することがわかった．

1 はじめに

大規模言語モデル (LLM) はさまざまなベンチマークで専門家と同等，またはそれ以上の性能を示しており，その実用性の高さから開発が加速している [1]．LLM 開発の原動力となっているのは，モデルの良さを表す交差エントロピー損失がモデルサイズ，データセットサイズ，学習時の投入計算量に応じてべき乗則的減少を示すスケール則という経験則である [2]．スケール則に従い開発が加速した結果，現在の最高性能のモデルは数 100B パラメータを超えるほど大規模化しており，事前学習単体が一大事業となるほどコストが巨大化している．LLM の事前学習において問題となるのは，学習途中で損失が突然発散するロススパイクと呼ばれる現象である．ロススパイク発生後のモデル学習は異常な経過を辿ることが多く，最終的に事前学習が失敗する可能性が高い [3, 4]．ロススパイクはモデルサイズが大きくなるほど発生しやすいことが実験的に確認されている [3, 5]．大きなモデルほど事前学習の時間的，金銭的コストが大きくなるため，ロススパイクの原理解明は，効率的な研究開発を進める上で重要である．

このような背景により，ロススパイクの発生を抑制する研究が盛んに行われている [3, 6, 7]．主なアプローチとしては，ロススパイクの原因を特定し，その要因を緩和する方法が挙げられる．現在までに，ロススパイクの原因として，学習データ中のノイズ [5]，コンテキストサイズ [8]，パラメータのノルムの不均一性 [6]，アテンションロジットの極端な増大 [7]，勾配ノルムの急激な増大 [9] などが指摘されている．

一方，ロススパイクがモデルに与える影響については十分に理解されていない．例えば，ロススパイクが発生した場合でもその後ロスが元に戻り学習を継続する場合があるが，このときスパイクの影響がモデルにどの程度残るのかは重要な疑問である．

本稿では，ロススパイクの影響を線形峰接続性 [10, 11] の観点から分析する．線形峰接続は，2 つの解が損失地形においてどのように関連しているかを表す．本稿では 2 つの解として，同一モデルの学習過程で生じる 2 つのチェックポイントを用いることで，学習中のモデルの挙動を観察する．実験では，Transformer デコーダベースの小規模な言語モデルを事前学習し，ロススパイクが発生する設定と発生しない設定における学習中のチェックポイントを記録・分析した．実験の結果，パラメータがスパイクの前後で大きく異なる位置に変化すること，及び線形峰接続性のパターンが変化することが確認された．

2 線形峰接続性

深層学習モデルの汎化性能の評価手法として，損失地形の平坦さを観察する手法がある [12, 13, 14, 15]．これは周囲が平坦な空間上のパラメータほど多少の誤差に対して安定しているという考察によるものだが，パラメータ次元が非常に大きい深層学習モデルでは，損失地形を正確に把握した

り可視化したりすることは容易ではない.

これに対し, 線形峰接続性は損失地形の一部を見るシンプルな概念であり, 損失地形における 2 地点の結びつきの強さ表す概念である. 2 つの地点 (パラメータ) $\theta_A, \theta_B \in \mathbb{R}^d$ が線形峰接続されているとは, 任意の $\alpha \in [0,1]$ について, 次の条件を満たすことを指す. ただし d はモデルのパラメータ数.

$$L((1-\alpha)\theta_A + \alpha\theta_B) \leq (1-\alpha)L(\theta_A) + \alpha L(\theta_B) \quad (1)$$

この条件は, θ_A と θ_B の線分上の任意の点において損失が低いままであることを意味する.

このように 2 つの地点の間の損失地形に限定することで分析を容易にし, 様々な知見が得られてきた. 例として, 線形接続性を非線形な経路に拡張した峰接続性を用いて, 異なる初期値から勾配降下法で得られた局所最小値同士でも, 多くの場合低損失の経路で接続することが, 経路を探索する方法とともに実験的に示された [10, 11]. また, Transformer エンコーダモデルのファインチューニングの分析に応用されている [16].

3 事前学習による分析対象の獲得

本実験では, 小規模な Transformer デコーダベースの言語モデルをロススパイクが発生する設定を含む複数の学習設定で事前学習し, チェックポイントを用いてロススパイクの影響を分析した.

言語モデルとして Llama-3.2-1B[1] を用いた. 学習時のコンテキストサイズは 512 とし, FineWeb-Edu [17] の一部を用いて学習を行った. FineWeb-Edu は, Common Crawl から収集された英語データセットである FineWeb [18] のうち, 教育的なコンテンツを抽出したサブセットである. 最適化アルゴリズムには AdamW を用い, ハイパーパラメータは $\beta_1 = 0.9$, $\beta_2 = 0.999$, $\epsilon = 1e-8$ とした. 最大学習率及び勾配クリッピングは表 1 のように複数の設定を用いた. 最大ステップ数は 35,075 とし, 学習率スケジューリングに 1,000 ステップのウォームアップとコサイン減衰を採用した. 最小学習率は 1e-5 とした. バッチサイズは 512 とした. 学習用計算機には mdx クラスタ上の NVIDIA A100 40GB GPU を用いた. 学習スクリプトは HuggingFace Transformers[2] を参考に, PyTorch を用いて実装した. モデルパラメータの初期値および乱数シードは全設定で統一した. 事前学習中のモデルの分析のために, チェックポイントを 50 ステップごとに保存した. チェックポイントあたりのデータ量は 4.7GB であった.

1) https://huggingface.co/meta-llama/Llama-3.2-1B
2) https://github.com/huggingface/transformers

表 1 学習設定

	(1)	(2)	(3)
最大学習率	1e-3	1e-3	5e-4
勾配クリッピング	–	1.0	–
ロススパイク発生有無	✓	✗	✗

4 分析結果

学習中の損失の過程 表 1 で定義した各設定における事前学習の損失の過程を図 4 に示す. 設定 (1) の場合, ロススパイクが 400 ステップ付近で起きたが, すぐ元に戻って学習が進んだ. 設定 (2), (3) においては, ロススパイクは生じず滑らかに学習が進んだ. これは, 学習率が大きいほどロススパイクが生じやすいという先行研究の結果と一致する [19].

チェックポイント間の損失地形 ここでは, 学習設定 (1), (3) における, チェックポイント間の線分上の損失地形を観察する. 損失計算には, FineWeb-Edu の先頭 100 事例を用いた. 結果を図 11 に示す. 設定 (1) の場合は, ロススパイク前後のチェックポイントのペア $(\theta_A, \theta_B) = (350, 450)$ において損失の山があることがわかる. 一方, 設定 (3) の場合は, どのペアにおいても損失の山が存在せず, 滑らかに接続されていることがわかる.

チェックポイント間の線形峰接続性 各学習設定において, 学習中のチェックポイント間の線形峰接続性の関係を調べるために, 50 ステップから 950 ステップまでのチェックポイント全ての組み合わせについて線形峰接続性を計算し, グラフで表した. グラフの点は各チェックポイントを表し, チェックポイント間で線形峰接続性の条件が満たされていれば辺で結ぶものとする. 各設定におけるグラフを図 15 に示す. ロススパイクが発生した設定 (1) においては, ロススパイク付近の 400 ステップで接続性が分断していることがわかる. 一方, ロススパイクが発生しなかった設定 (2) や (3) の場合は, ほとんどの場合において, 隣り合うチェックポイント同士は辺で繋がっており, 特に学習後期のチェックポイント間は密に接続していることがわかる. このことから, ロススパイクが発生しない, 即ち損失が滑らかに降下するとき, チェックポイント同士の接続性が保たれることがわかる.

チェックポイント間のパラメータの距離 チェッ

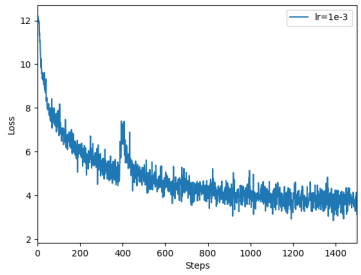

図 1　LR=1e-3

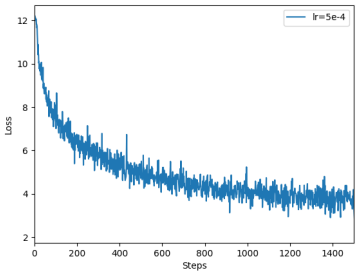

図 2　LR=5e-4

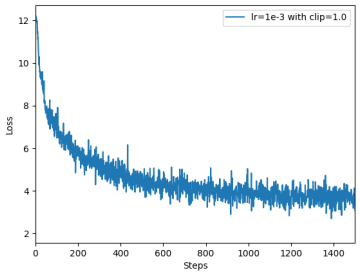

図 3　LR=1e-3 with clip = 1.0

図 4　損失の過程.

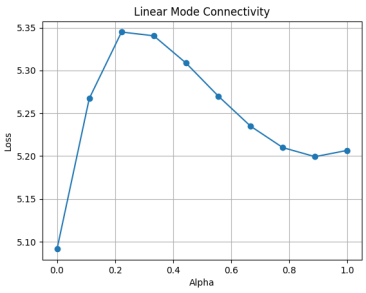

図 5　$(\theta_A, \theta_B) = (350, 450)$ (LR=1e-3).

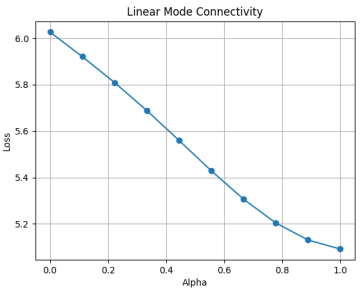

図 6　$(\theta_A, \theta_B) = (200, 350)$ (LR=1e-3).

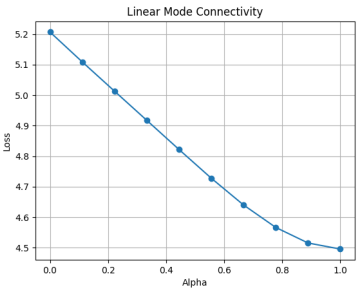

図 7　$(\theta_A, \theta_B) = (450, 600)$ (LR=1e-3).

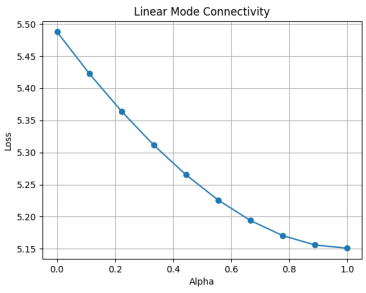

図 8　$(\theta_A, \theta_B) = (350, 450)$ (LR=5e-4).

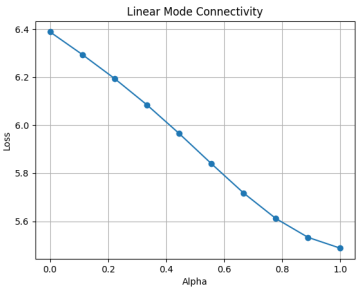

図 9　$(\theta_A, \theta_B) = (200, 350)$ (LR=5e-4).

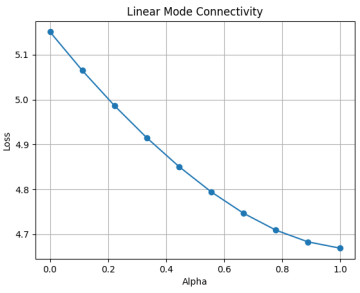

図 10　$(\theta_A, \theta_B) = (450, 600)$ (LR=5e-4).

図 11　2 つのパラメータの線分上の損失地形. ($\alpha = \{0, 1/9, .., 8/9, 1.0\}$)

クポイント間の線形峰接続を見るだけでは, パラメータ同士の近さを損失の観点でしか測ることができない. そこで, チェックポイント間のパラメータの距離を測定した. ここでは距離としてユークリッド距離を用いた. 結果を図 19 に示す. ステップ差が大きいほどパラメータ同士のの距離が大きいことがわかる. また, 学習初期よりも学習後期のチェックポイント同士の距離の方が大きいことがわかる. 設定 (1) の場合に着目すると, ロススパイク発生付近の 400 ステップの前後で距離が大きく変化していることがわかる. これは, ロススパイクによってパラメータが大きく変化したためと考えられる.

5　おわりに

本稿では, ロススパイクの影響を線形峰接続性の観点から分析した. 小規模な言語モデルをロススパイクが発生する設定を含む複数の学習設定で事前学習し, チェックポイントを用いてスパイクの影響を分析した. その結果, スパイクの発生前後でパラメータが大きく異なる位置にずれること, 線形峰接続のパターンが異なることがわかった. なお, スパイク発生後のモデルの挙動は, スパイク直前の損失に戻る場合, 完全に発散してしまう場合など様々なパターンがあるが, 各パターンの影響の違いについては今後の調査が必要である. また, 本研究では 1.3B という比較的小さいサイズのモデルを扱ったが, より大きいモデルについても調査が必要である.

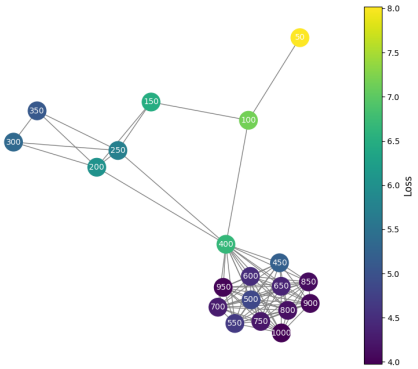

図 12　LR=1e-3

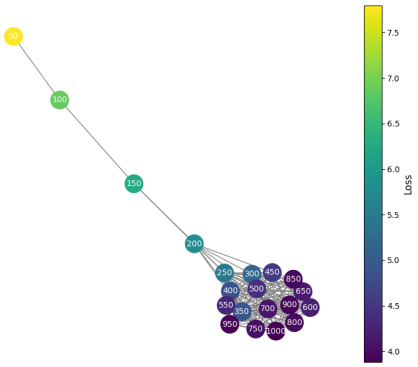

図 13　LR=1e-3 with clip = 1.0

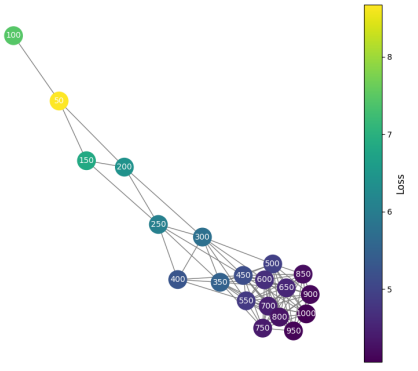

図 14　LR=5e-4

図 15　チェックポイント間の線形峰接続性をグラフで図示したもの. グラフの点上の数字はチェックポイントのステップ数を表す. 点の濃淡はチェックポイントにおける損失を表す.

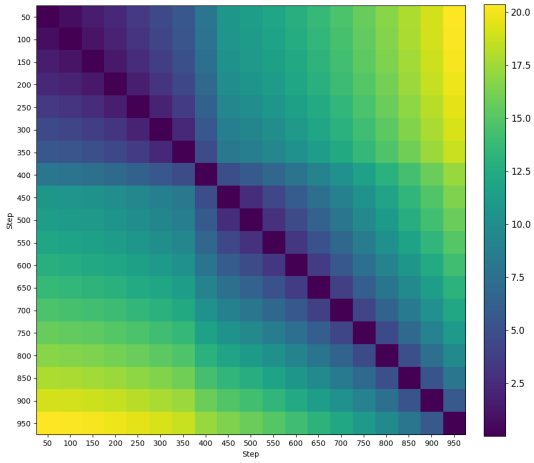

図 16　LR=1e-3

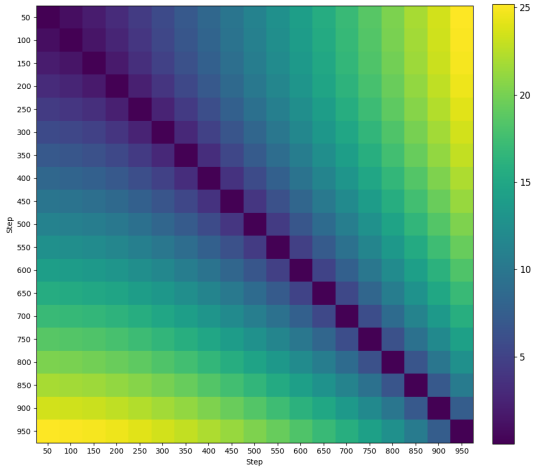

図 17　LR=1e-3 with clip = 1.0

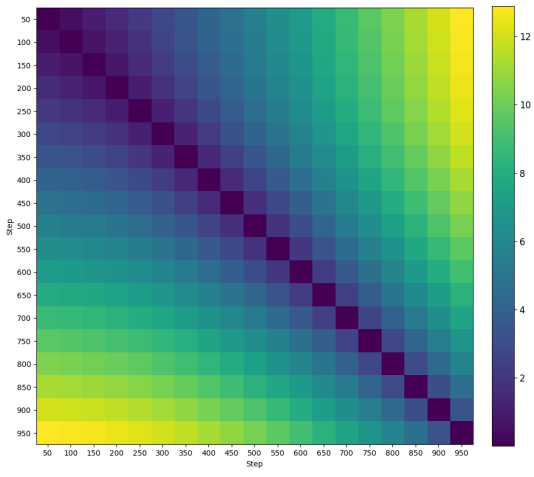

図 18　LR=5e-4

図 19　チェックポイント間のパラメータの距離行列.

謝辞

著者の杉浦は, 2024 年度公益財団岩垂奨学会から奨学金を受給しました.

参考文献

[1] Wayne Xin Zhao, Kun Zhou, Junyi Li, Tianyi Tang, Xiaolei Wang, Yupeng Hou, Yingqian Min, Beichen Zhang, Junjie Zhang, Zican Dong, et al. A survey of large language models. **arXiv preprint arXiv:2303.18223**, 2023.

[2] 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.

[3] Aakanksha Chowdhery, Sharan Narang, Jacob Devlin, Maarten Bosma, Gaurav Mishra, Adam Roberts, Paul Barham, Hyung Won Chung, Charles Sutton, Sebastian Gehrmann, et al. Palm: Scaling language modeling with pathways. **Journal of Machine Learning Research**, 2023.

[4] Susan Zhang, Stephen Roller, Naman Goyal, Mikel Artetxe, Moya Chen, Shuohui Chen, Christopher Dewan, Mona Diab, Xian Li, Xi Victoria Lin, et al. Opt: Open pre-trained transformer language models. **arXiv preprint arXiv:2205.01068**, 2022.

[5] Team OLMo, Pete Walsh, Luca Soldaini, Dirk Groeneveld, Kyle Lo, Shane Arora, Akshita Bhagia, Yuling Gu, Shengyi Huang, Matt Jordan, Nathan Lambert, Dustin Schwenk, Oyvind Tafjord, Taira Anderson, David Atkinson, Faeze Brahman, Christopher Clark, Pradeep Dasigi, Nouha Dziri, Michal Guerquin, Hamish Ivison, Pang Wei Koh, Jiacheng Liu, Saumya Malik, William Merrill, Lester James V. Miranda, Jacob Morrison, Tyler Murray, Crystal Nam, Valentina Pyatkin, Aman Rangapur, Michael Schmitz, Sam Skjonsberg, David Wadden, Christopher Wilhelm, Michael Wilson, Luke Zettlemoyer, Ali Farhadi, Noah A. Smith, and Hannaneh Hajishirzi. 2 olmo 2 furious. **arXiv preprint arXiv:2501.00656**, 2024.

[6] Kosuke Nishida, Kyosuke Nishida, and Kuniko Saito. Initialization of large language models via reparameterization to mitigate loss spikes. In **EMNLP**, 2024.

[7] Mostafa Dehghani, Josip Djolonga, Basil Mustafa, Piotr Padlewski, Jonathan Heek, Justin Gilmer, Andreas Peter Steiner, Mathilde Caron, Robert Geirhos, Ibrahim Alabdulmohsin, Rodolphe Jenatton, Lucas Beyer, Michael Tschannen, Anurag Arnab, Xiao Wang, Carlos Riquelme Ruiz, Matthias Minderer, Joan Puigcerver, Utku Evci, Manoj Kumar, Sjoerd Van Steenkiste, Gamaleldin Fathy Elsayed, Aravindh Mahendran, Fisher Yu, Avital Oliver, Fantine Huot, Jasmijn Bastings, Mark Collier, Alexey A. Gritsenko, Vighnesh Birodkar, Cristina Nader Vasconcelos, Yi Tay, Thomas Mensink, Alexander Kolesnikov, Filip Pavetic, Dustin Tran, Thomas Kipf, Mario Lucic, Xiaohua Zhai, Daniel Keysers, Jeremiah J. Harmsen, and Neil Houlsby. Scaling vision transformers to 22 billion parameters. In **ICML**, 2023.

[8] Conglong Li, Minjia Zhang, and Yuxiong He. The stability-efficiency dilemma: Investigating sequence length warmup for training gpt models. **NeurIPS**, 2022.

[9] Sho Takase, Shun Kiyono, Sosuke Kobayashi, and Jun Suzuki. Spike no more: Stabilizing the pre-training of large language models. **arXiv preprint arXiv:2312.16903**, 2023.

[10] Timur Garipov, Pavel Izmailov, Dmitrii Podoprikhin, Dmitry P Vetrov, and Andrew G Wilson. Loss surfaces, mode connectivity, and fast ensembling of dnns. In **NeurIPS**, 2018.

[11] Felix Draxler, Kambis Veschgini, Manfred Salmhofer, and Fred Hamprecht. Essentially no barriers in neural network energy landscape. In **ICML**, 2018.

[12] Sepp Hochreiter and Jürgen Schmidhuber. Flat minima. **Neural Comput.**, 1997.

[13] Nitish Shirish Keskar, Dheevatsa Mudigere, Jorge Nocedal, Mikhail Smelyanskiy, and Ping Tak Peter Tang. On large-batch training for deep learning: Generalization gap and sharp minima. **arXiv preprint arXiv:1609.04836**, 2016.

[14] 佐藤竜馬. 深層ニューラルネットワークの高速化. 技術評論社, 2024.

[15] Hao Li, Zheng Xu, Gavin Taylor, Christoph Studer, and Tom Goldstein. Visualizing the loss landscape of neural nets. In **NeurIPS**, 2018.

[16] Yujia Qin, Cheng Qian, Jing Yi, Weize Chen, Yankai Lin, Xu Han, Zhiyuan Liu, Maosong Sun, and Jie Zhou. Exploring mode connectivity for pre-trained language models. In **EMNLP**. Association for Computational Linguistics, 2022.

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

[18] Guilherme Penedo, Hynek Kydlíček, Loubna Ben allal, Anton Lozhkov, Margaret Mitchell, Colin Raffel, Leandro Von Werra, and Thomas Wolf. The fineweb datasets: Decanting the web for the finest text data at scale. In **NeurIPS Datasets and Benchmarks Track**, 2024.

[19] Mitchell Wortsman, Peter J Liu, Lechao Xiao, Katie Everett, Alex Alemi, Ben Adlam, John D Co-Reyes, Izzeddin Gur, Abhishek Kumar, Roman Novak, et al. Small-scale proxies for large-scale transformer training instabilities. **arXiv preprint arXiv:2309.14322**, 2023.