文脈の記憶と想起を行う状態空間モデルによる状態遷移の分析

山本悠士　松崎拓也
東京理科大学大学院 理学研究科 応用数学専攻
1423531@ed.tus.ac.jp　matuzaki@rs.tus.ac.jp

概要

Attention は，文章生成時のメモリ消費量が入力トークン数に比例して増加するため，長文の処理や高速な文章生成を困難にする．この課題を解決するため，推論時のメモリサイズが入力長によらず一定である状態空間モデル (SSM) に基づいた言語モデルが提案されている．本研究は，SSM に基づく言語モデルがどのようにして入力文中の情報を記憶しているのかを分析する．分析では，記憶と想起を必要とする質問応答タスクにおいて，正答に必要不可欠な SSM をもつ層を特定して，その SSM が本文中のどの文脈を参照しているかを観察する．記憶と想起のメカニズムの実現のために SSM のパラメータが満たすべき条件を示すことは今後の課題とする．

1　はじめに

現在，自然言語処理分野では Transformer ベースの言語モデルが広く活用されているが，Transformer と同程度の性能を達成しつつより効率的に計算できるアーキテクチャの開発や分析も盛んに行われている．Transformer の主要モジュールである Attention [1] は，文章生成時に過去に入力されたトークンに対応する中間表現を保持する必要があるため，生成時のメモリ消費量が入力トークン数に比例して増加する．この性質は長文の処理や高速な文章生成を困難にする．この課題を解決するため，推論時のメモリ消費量が入力トークン数に依存しない Linear Attention や状態空間モデル (State Space Model; SSM) などが提案され，改善が続けられている [2, 3]．

推論時のメモリサイズが入力長によらず一定である Mamba [4] などの言語モデルは，どのようにして入力文中の情報を記憶しているのか？ Mamba は SSM ベースのモデルであり，本研究の主な分析対象である．SSM は，RNN と同様に，文章を 1 トークンずつ受け取り，記憶に対応する隠れ状態を再帰的に更新することで文脈を捉える．本研究で使用した Mamba がもつ SSM の隠れ状態の要素数は，SSM への入力の 16 倍である．つまり，SSM は文章内の情報を 16 トークン分に対応する表現に圧縮しなければならない．そのため，SSM が単語間の長距離依存を捉えるためには，重要なトークンを厳選するなどのメモリ効率のよい推論を行う必要がある．

本研究では，質問応答タスクを題材として，SSM が回答に必要な本文中の情報を記憶するメカニズムを分析する．本文の内容に関する質問が本文に続けて与えられる場合，SSM はまず本文の要点を記憶し，その後，質問に対応する文脈を想起することが求められる．現時点では，この記憶と想起のメカニズムを実現するために SSM のパラメータが満たすべき条件の解明には至っていない．この原因は，多層モデルである Mamba には SSM モジュールが複数存在し，それぞれが多様な推論を行うことで分析対象の成分が複雑化するためである．そこで，まず，記憶と想起を必要とする質問応答タスクにおいて，正答に必要不可欠な SSM をもつ層の特定を行う (§5.1)．その後，特定された層にある SSM が本文中のどの文脈を参照しているかを観察する (§5.2)．特定された一部の SSM のパラメータの共通性に基づいて，記憶と想起を可能とするメカニズムを具体的に示すことは今後の課題とする．

2　関連研究

我々はこれまでに，帰納ヘッドタスク（induction heads task）[5] のもとで，SSM が過去の文脈に基づいて推論を行うメカニズムを分析してきた [6]．帰納ヘッドタスクとは，入力文の末尾と似ている文脈を過去の入力から参照する状況を抽象化した合成タスクである．具体的には，abcde. . . c というトークン列が入力された場合は，末尾のトークン c が前回出現した位置の次のトークンである d を出力するタスクである．我々は，帰納ヘッドタスクにおいて，1 層の Mamba が過去の文脈を参照するメカニズムを実現するためのパラメータの候補を示した [6]．

しかし，帰納ヘッドタスクの入力はランダムなトークン列であるため，自然言語との乖離が大きい．そこで本研究では，問題設定を質疑応答タスクである SQuAD [7] に拡張し，分析対象のモデルもより実用的な 24 層の事前学習済みの Mamba に変更する．

モデル構成によって言語現象の認識力にどれだけ差が生じるかを推測するため，特定の言語現象を抽象化した合成タスクによる評価が行われている．Olsson ら [5] は，Transformer による文脈内学習の大半は帰納ヘッドタスクが実行可能な Attention ヘッドにより実現されることを示唆した．また，Arora ら [8] は，Attention をもたないモデルの言語モデリング性能が Attention をもつモデルよりも低い原因が，文脈中で既に言及された情報を想起する連想想起能力が低いためであると説明した．このような背景から，新しい言語モデルを設計する際は，過去の文脈を想起する能力に焦点が当てられている．

Sharma ら [9] は，Mamba が事前学習時に記憶した知識を想起する際，過去の文脈から連想される情報がモデルの後半の層にある SSM を介して最後のトークンに伝達されることを示した．これは，文脈内の情報を記憶して想起する質問応答においてモデルの後半に位置する SSM は必要不可欠だったという本稿の §5.1 の結果と類似している．

3 準備：状態空間モデル

本節では，言語モデルである Mamba の主要モジュールである状態空間モデル（SSM）について説明する．本稿では，入出力 $\boldsymbol{x}, \boldsymbol{y} \in \mathbb{R}^D$ に対する SSM の処理を次のように表す：

$$H_t = \overline{A}_t \odot H_{t-1} + \overline{B}_t \odot (\mathbb{1}_N \boldsymbol{x}_t^\top) \qquad \in \mathbb{R}^{N \times D} \tag{1}$$

$$\boldsymbol{y}_t^\top = \boldsymbol{c}_t^\top H_t + (\boldsymbol{d} \odot \boldsymbol{x}_t)^\top. \tag{2}$$

以降，H_t を**隠れ状態**と呼び，隠れ状態 H_t の行数 N を**状態サイズ**と呼ぶ．また，隠れ状態の初期値 H_{-1} はゼロ行列とする．隠れ状態 H_{t-1} と入力 $\boldsymbol{x}_t$ に対する係数 $\overline{A}_t, \overline{B}_t, \boldsymbol{c}_t$ は次のように定義される：

$$\overline{A}_t = \exp(-\Delta_t \odot A),\ \overline{B}_t = \Delta_t \odot (W_B \boldsymbol{x}_t \mathbb{1}_D^\top),\ \boldsymbol{c}_t = W_c \boldsymbol{x}_t,$$

$$\Delta_t = \mathbb{1}_N\, (\mathrm{softplus}(W_\Delta \boldsymbol{x}_t + \boldsymbol{b}_\Delta))^\top$$

ただし，$A, \boldsymbol{d}, W_\square, \boldsymbol{b}_\square$ は学習パラメータである．

Mamba は，式 1-2 で定義された SSM に加えて，畳み込み層と活性化関数である SwiGLU [10] から構成された Mamba 層が複数積み重なった言語モデルである．本稿では，SSM のみに注目するため，Mamba 全体の構成についての詳説は文献 [4] に委ねる．

Title: Super_Bowl_50
*Background: Six-time Grammy winner and Academy Award nominee **Lady Gaga** performed the national anthem, while Academy Award winner Marlee Matlin provided American Sign Language (ASL) translation.*
Question: Who sang the national anthem?
*Answer: **Lady Gaga***

図 1 SQuAD のサンプルを文章化した例．SQuAD は，タイトル，本文，質問，正答の 4 つのフィールドからなる構造化データであるため，各項目にプレフィックスを付け，それらを結合することでテキスト化する．回答生成時は「*Answer:*」までを入力する．

4 実験設定

本節では，分析対象である質問応答タスクの設定と使用したモデルについて説明する．

4.1 タスク設定

本研究では，質問応答タスクである SQuAD [7] [1] を題材として，モデルが文脈の記憶と想起を行うメカニズムを分析する．SQuAD は，Wikipedia の記事の抜粋である本文と質問応答ペアから構成されるデータセットである．モデルに本文と質問文を入力して回答を出力させる手順は，LM Evaluation Harness [11] と同一とした．すなわち，モデルへの入力は図 1 のフォーマットに基づいて作成し，回答は自己回帰によるトークン生成によって行う．

4.2 モデルとファインチューニング

分析には Gu ら [4] が公開している事前学習済みの Mamba (130M) を用いる．[2] Mamba (130M) の層数は 24 であり，SSM の状態サイズ N は 16 である．また，文脈の記憶と想起をすることで質問応答を実行するメカニズムを分析するためには，モデルが質問応答を行う能力を持っていることが前提となるため，SQuAD の訓練データ（1 エポック）を用いてファインチューニングを行う．ファインチューニングする際の損失は，回答として生成されたトークン列と正答に対するクロスエントロピー損失とした．

Mamba を学習する前後での検証データにおける SQuAD のスコアを表 1 に示す．学習によりスコアが格段に向上していることが分かる．54.2% という正解率（完全一致率）は，語彙サイズが約 5 万の言語モデルが次単語予測によって回答を生成していることを考慮すれば，十分高いスコアである．した

1） https://huggingface.co/datasets/rajpurkar/squad
2） https://huggingface.co/state-spaces/mamba-130m-hf

表 1　ファインチューニング前後での SQuAD のスコア．スコアの測定前にモデルの出力と正答の先頭と末尾からピリオド・カンマ・スペースを削除した．

	正解率 (%)	適合率 (%)	再現率 (%)	出力長 (トークン)
FT 前	3.28	26.0±31.9	8.31±20.9	101±108
FT 後	54.2	66.4±43.1	66.4±43.0	4.39±4.01

がって，学習後のモデルは文脈の記憶と想起を行うメカニズムを獲得していると考えられる．

5　分析：記憶と想起のメカニズム

本節では，文脈の記憶と想起を実現している SSM の推論メカニズムを分析した結果を述べる．まず，各層の SSM を無効化したときの生成結果の変化から，文章読解や文脈記憶に大きく寄与している隠れ状態を特定する (§5.1)．次に，隠れ状態の遷移を可視化することで特定された隠れ状態が本文中のどの文脈を捉えているかを観察する (§5.2)．最後に，隠れ状態の分析をより客観的に行うために独立成分分析の活用を検討する (§5.3)．

5.1　質問応答に不可欠な SSM の特定

SQuAD の検証データにおいて，ファインチューニング後のモデルでは正答したが，特定の層の SSM を無効化すると不正解となる状況を考える．有効化しないと質問応答に失敗するような SSM は，文章中の正解のキーワードに関わる文脈の記憶や質問文の理解などの重要な役割を担っている可能性が高い．ここで SSM を無効化するとは，隠れ状態を更新しないことで常に $H_t = O$ とすることを意味する．

分析には SQuAD の検証データにおいてファインチューニング後の Mamba が正答した 1,768 件を用いた．SSM の機能の違いを調査するために，どのカテゴリを回答すべき問題であるかに基づいて各サンプルを分類した．カテゴリは，人名・人名以外の固有名詞・数量・日付の 4 種類であり，分類は質問文の疑問詞を基準に機械的に行った．例えば，疑問詞が When である問題には日付のカテゴリを付与した．

各層の SSM を無効化する前後で尤度がどれだけ低下するかを計算して，質問応答の正答に必要不可欠な SSM をもつ層を特定した．すなわち，無効化前の完全なモデルの尤度 $P(A|X,Q)$ と第 ℓ 層の SSM を無効化したモデルの尤度 $P^{\times}_{\ell}(A|X,Q)$ との差

$$d^{\times}_{\ell} = P(A|X,Q) - P^{\times}_{\ell}(A|X,Q) \tag{3}$$

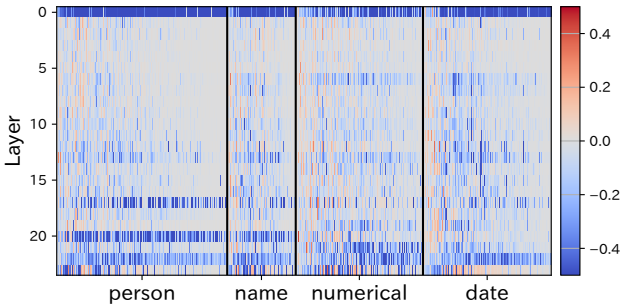

図 2　各層の SSM を無効化したときの尤度の変化．

を計算した．ただし，X, Q, A はそれぞれ各サンプルの本文，質問文，正答である．尤度差 $d^{\times}_{\ell}$ $(\ell = 1, \ldots, L)$ を各サンプルについて計算した結果を図 2 に示す．全体的に尤度差はゼロ付近のものが多い．すなわち，1 つの層を無効化しても尤度は変化しない場合が多い．しかし，第 0 層[3)]を無効化すると尤度が平均で約 0.61 低下する．これは，Mamba が第 0 層においてモデル全体の処理に必要不可欠な表層的な処理を行っているためだと推測される．また，第 17 層と第 20 層を無効化すると固有名詞を回答すべき問題で尤度が低下する傾向がある．これより，文章中のキーワードの記憶または想起には，第 17 層と第 20 層の SSM が行う状態遷移が必要不可欠であると考えられる．

5.2　隠れ表現に保持された本文の記憶

前項では，文章読解や文脈記憶に寄与している SSM をもつ層を特定したが，その SSM の推論メカニズムは不明なままである．そこで本項では，回答の直前のトークン *Answer:* が入力された直後の隠れ状態に文章中のどの情報が記憶されているかを分析する．具体的には，各トークンの隠れ状態 H_t と回答直前の隠れ状態 H_T の各行の内積を計算する．

図 1 の例文全体をモデルに入力したとき，SSM の隠れ状態 H の各行には何の情報が記憶されるのか．ファインチューニング後のモデルに例文を入力すると正答である *Lady Gaga* が生成されるが，第 19 層の SSM を無効化すると正答に対する尤度が約 0.14 低下し，誤答である *Marlee Matlin* が生成される．特に，第 19 層にある SSM の隠れ状態 $H_t \in \mathbb{R}^{16\times D}$ を行ごとに無効化したときに尤度の減少幅が大きい上位 4 行（0,9,11,15 行目）のみを同時に無効化した場合も *Marlee Matlin* と誤答する．このような質問応答に正答するために不可欠な隠れ状態の成分が入力中のどの文脈に依存しているのかを調べるため，第

3)　本稿では，層や隠れ状態の行数を 0 を起点として数える．

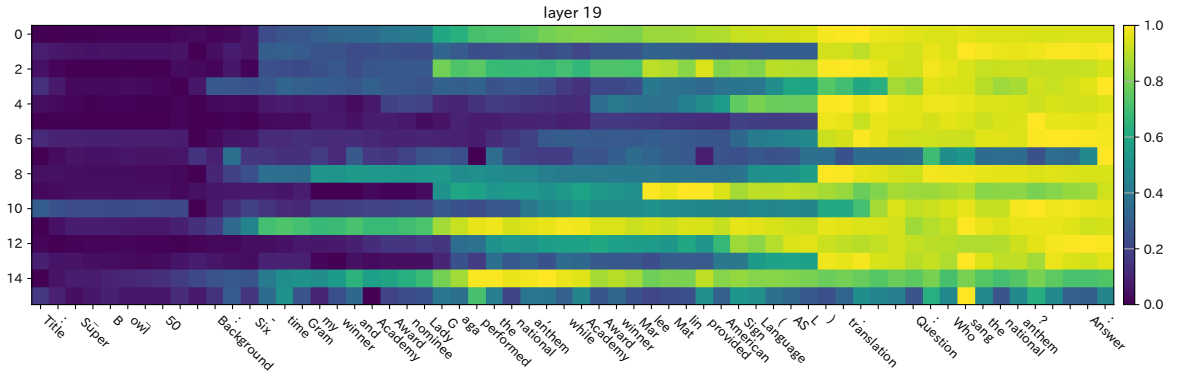

図 3　第 19 層の SSM における，各トークンに対応する隠れ状態 H_t と末尾トークンに対応する隠れ状態 H_T の内積．このときの入力は，図 1 の例文である．可視化する際に，行ごとに min-max 正規化を適用している．

表 2　ICA 適用後の埋め込みについて，第 d 次元の成分が上位 5 位となるトークン．

次元 d	解釈	上位 5 位のトークン
50	時間	years, months, decades, weeks, year
58	800 台	893, 873, 892, 882, 872
170	欧州	france, spain, italy, germany, portugal
319	数量	amount, number, quantity, size, length

19 層の SSM について，各隠れ状態 H_t と最後の隠れ状態 H_T の各行の内積を計算した．図 3 に示した内積の変化によると，0,2,9,11,14 行目は *Lady Gaga* 付近で類似度が急上昇していることが分かる．この内，9 行目は *Marlee Matlin* 付近でより類似度が高くなる．従って，これらの行は本文中の人名を記憶を保持していることが示唆される．また，15 行目は質問文内の *sang* で類似度が最大となることから，2 人のうち歌手の方を回答することに寄与していることが示唆される．

5.3　埋め込みと隠れ状態に対する ICA

本項では，SSM がどのカテゴリの情報を記憶しているのかを客観的に分析する手法を検討する．前項で隠れ状態に本文中のどの情報が記憶されているかを観察したが，この可視化に基づく手順は主観的であり，人手を要するため多様なサンプルに対して網羅的に分析することが難しい．そこで，埋め込み空間におけるカテゴリの軸を発見することができる独立成分分析（ICA）の利用を検討する [12].

Mamba のトークン埋め込み行列に対して ICA を適用し，変換後の埋め込みの各次元の成分が大きいトークンを列挙すると，同一のカテゴリに属するトークンが得られた．一部の次元に対する結果を表 2 に示す．この結果から，Mamba による国名，数量，時間に関する推論は，ICA で射影された軸のうち，それぞれ第 170 軸，第 319 軸と第 58 軸，第 50 軸の上で行われると考えられる．

しかし，現時点では，トークン埋め込みで学習された ICA モデルを用いて SSM の隠れ状態を解釈することはできていない．この原因は，隠れ状態がトークン単位ではなく第 0 層の受容野に対応する N-gram 単位の表現であることや，残差接続がある層の内部表現は直前の層との差分であってトークン自体を表現していないことなどが考えられる．

6　おわりに

本研究では，質問応答タスクを題材として，推論時のメモリサイズが一定という制約がある SSM がどのように過去の文脈を参照しているのかを分析した．まず，Mamba において過去の文脈を参照している SSM をもつ層を特定するために，各層の SSM を無効化することでそれぞれの質問応答に対する寄与を測った．その結果，質問応答に正答する際に必要不可欠な SSM があることが分かった．その後，隠れ状態の可視化を通じて，これらの SSM が文章中のどの文脈を参照しているかを観察した．

今後の課題は，文脈の記憶と想起を行っていることが示唆される SSM を対象に，記憶と想起を実現するために SSM が満たす必要があるパラメータの条件などを具体的に示すことである．また，本稿の限界として，今回用いた Mamba は SQuAD でファインチューニングされているため，ゼロショット推論時に同じ挙動を示すとは言えないことが挙げられる．また Linear Attention に基づくモデルは，式 1 を単純化した Mamba2 [13] や同式に記憶の消去を導入した DeltaNet [14, 15] など様々であり，記憶の更新規則の違いがモデルの推論にどのように寄与するのかも興味深い．

参考文献

[1] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, L ukasz Kaiser, and Illia Polosukhin. Attention is all you need. In I. Guyon, U. Von Luxburg, S. Bengio, H. Wallach, R. Fergus, S. Vishwanathan, and R. Garnett, editors, **Advances in Neural Information Processing Systems**, Vol. 30. Curran Associates, Inc., 2017.

[2] Songlin Yang and Yu Zhang. FLA: A triton-based library for hardware-efficient implementations of linear attention mechanism, January 2024. https://github.com/fla-org/flash-linear-attention.

[3] Xindi Wang, Mahsa Salmani, Parsa Omidi, Xiangyu Ren, Mehdi Rezagholizadeh, and Armaghan Eshaghi. Beyond the limits: A survey of techniques to extend the context length in large language models. In Kate Larson, editor, **Proceedings of the Thirty-Third International Joint Conference on Artificial Intelligence, IJCAI-24**, pp. 8299–8307. International Joint Conferences on Artificial Intelligence Organization, August 2024. Survey Track.

[4] Albert Gu and Tri Dao. Mamba: Linear-time sequence modeling with selective state spaces. In **First Conference on Language Modeling**, 2024.

[5] Catherine Olsson, Nelson Elhage, Neel Nanda, Nicholas Joseph, Nova DasSarma, Tom Henighan, Ben Mann, Amanda Askell, Yuntao Bai, Anna Chen, Tom Conerly, Dawn Drain, Deep Ganguli, Zac Hatfield-Dodds, Danny Hernandez, Scott Johnston, Andy Jones, Jackson Kernion, Liane Lovitt, Kamal Ndousse, Dario Amodei, Tom Brown, Jack Clark, Jared Kaplan, Sam McCandlish, and Chris Olah. In-context learning and induction heads. **Transformer Circuits Thread**, 2022. https://transformer-circuits.pub/2022/in-context-learning-and-induction-heads/index.html.

[6] 山本悠士, 松崎拓也. Mamba ブロックが帰納ヘッドタスクを実行するメカニズム. 情報処理学会研究報告 自然言語処理 (NL) , 2024-NL-260.

[7] Pranav Rajpurkar, Jian Zhang, Konstantin Lopyrev, and Percy Liang. SQuAD: 100,000+ questions for machine comprehension of text. In Jian Su, Kevin Duh, and Xavier Carreras, editors, **Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing**, pp. 2383–2392, Austin, Texas, November 2016. Association for Computational Linguistics.

[8] Simran Arora, Sabri Eyuboglu, Aman Timalsina, Isys Johnson, Michael Poli, James Zou, Atri Rudra, and Christopher Re. Zoology: Measuring and improving recall in efficient language models. In **The Twelfth International Conference on Learning Representations**, 2024.

[9] Arnab Sen Sharma, David Atkinson, and David Bau. Locating and editing factual associations in Mamba. In **First Conference on Language Modeling**, 2024.

[10] Noam Shazeer. GLU variants improve transformer. arXiv:2002.05202, 2020.

[11] 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, July 2024. https://zenodo.org/records/12608602.

[12] Hiroaki Yamagiwa, Momose Oyama, and Hidetoshi Shimodaira. Discovering universal geometry in embeddings with ICA. In Houda Bouamor, Juan Pino, and Kalika Bali, editors, **Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing**, pp. 4647–4675, Singapore, December 2023. Association for Computational Linguistics.

[13] Tri Dao and Albert Gu. Transformers are SSMs: Generalized models and efficient algorithms through structured state space duality. In **Forty-first International Conference on Machine Learning**, 2024.

[14] Imanol Schlag, Kazuki Irie, and Jürgen Schmidhuber. Linear transformers are secretly fast weight programmers. In Marina Meila and Tong Zhang, editors, **Proceedings of the 38th International Conference on Machine Learning**, Vol. 139 of **Proceedings of Machine Learning Research**, pp. 9355–9366. PMLR, 18–24 Jul 2021.

[15] Songlin Yang, Bailin Wang, Yu Zhang, Yikang Shen, and Yoon Kim. Parallelizing linear transformers with the delta rule over sequence length. In **The Thirty-eighth Annual Conference on Neural Information Processing Systems**, 2024.