定数精度浮動小数点 Transformer Decoder が認識する言語の有限性・余有限性

根岸直生 [1] 谷口雅弥 [2] 坂口慶祐 [1,2] 乾健太郎 [3,1,2]
[1] 東北大学 [2] 理化学研究所 [3]MBZUAI
naoki.negishi.s5@tohoku.ac.jp

概要

Transformer decoder の認識する言語を所属性問題として定義し，softmax 関数および定数精度浮動小数点数を採用した場合，認識する言語は有限言語および余有限言語クラスと一致することを示した．

1 はじめに

Transformer モデル [1] の表現力，すなわち本質的に解決可能な問題に関する近年の理論的解析によって，様々な仮定の下での表現力の上限・下限が明らかになりつつある [2]． 例えば 1 回のデコードに限定した場合，Transformer の表現力は回路複雑性クラス TC^0 程度である [3]． しかし思考の連鎖 [4] のような手法で多項式回のデコードを許容すると，その表現力が P にまで拡張されることが示されている．

一方で，これらの先行研究では理論的解析を容易にするため，行列計算に使用する浮動小数点数の精度について，無限精度 ($\mathbb{Q}$ や $\mathbb{R}$) や入力系列長に対して対数的に増加する精度（$\mathcal{O}(\log n)$ 精度）を仮定している． 実際のモデルは定数精度 (fp32 や bf16 等) で実装されており，十分に長い入力文字列の全情報を単一のベクトルに圧縮して保持できない． そのため，先行研究における枠組みでは実際のモデルの表現力を厳密に評価することは困難である． 更に，先行研究では Attention 機構において softmax 関数の代わりにしばしば hardmax 関数という理想化を行うが，両者は入力に対する出力の応答が大きく異なるため互いを完全に置換できない． 本研究では，**浮動小数点演算における定数精度の制約**および **softmax 関数の利用**を前提に，現実的な設定における Transformer decoder モデルの表現力を分析する．

本研究の結果として query と key の積に関する自然な仮定（仮定 9）の下で Transformer が認識可能な言語が有限言語または余有限言語（3.1 節）の対応が得られた（定理 10，表 1 左下）． さらに表 1 (1), (4) の灰色の場合については，本研究では直接扱わないが，定理 10 を拡張することで証明可能であると考える．

表 1 Transformer モデルの表現力の上限．(1),(3) は下限と等しい． (2) が先行研究 [3] による結果，(3) は本研究で証明した結果（定理 10）． (2), (4) はデコード回数を $\mathcal{O}(1), \mathcal{O}(\log n), \mathcal{O}(n)$ とした場合の上限をカンマで区切っている． また灰色で塗りつぶされている (1), (4) は本研究の定理を拡張することで証明可能と予想される結果．

	浮動小数点数精度	
	$\mathcal{O}(1)$, 仮定 9	$\mathcal{O}(\log n)$
hardmax	(1) Finite ∪ Co-finite? (定理 10 の応用)	(2) $\mathsf{TC}^0, \mathsf{L}, \mathsf{NC}^1$ [3]
softmax	(3) **Finite ∪ Co-finite (定理 10)**	(4) $\mathsf{TC}^0, \mathsf{L}, \mathsf{NC}^1$ (定理 10 の応用)

2 関連研究

ニューラルネットワークの表現力は，関数近似や計算複雑性によって特徴づけられてきた． 前者では，任意の連続関数を近似的に模倣できることを主張する万能近似定理が代表的な成果として挙げられる [5, 6]． 一方，後者では形式言語や論理回路等との対応を通じて表現力が評価され，整理されたサーベイ論文 [2] や講義ノート [7] が公開されている．

Transformer モデルの表現力については，理論的な制約がほぼ存在しない場合においてチューリング完全性が示されている [8]． さらに入力系列長に対して，モデル次元や出力系列長等を決定する関数のオーダーによる表現力の階層づけもされている [9, 3]． しかし現実の実装ではこのような仮定はしばしば破られることを考慮する必要がある．

3 準備

4 節に先立ち，分析で重要な役割を果たす諸概念の定義を行う． なおアルファベット Σ 上の文字列

$w, w' \in \Sigma^*$ について $|w|$ は文字列の長さを，w_t は t 番目の文字，また ww' は文字列の連接を表す．

3.1 有限言語・余有限言語

まず**有限言語**とその双対である**余有限言語**を定義し，これらの言語を認識する決定性有限オートマトン (DFA) を構築するためのアルゴリズムを示す．

定義 1 (有限言語)**.** アルファベット Σ 上の言語 L が**有限言語**であるとは，$\exists k \in \mathbb{N}. \forall w \in L. |w| \leq k$ が成立する場合，またその場合に限る．

定義 2 (余有限言語)**.** アルファベット Σ 上の言語 L が**余有限言語**であるとは，L の補集合 $\Sigma^* \setminus L$ が有限言語である場合，またその場合に限る．

例 3. 以下の $\Sigma = \{a, b\}$ 上の言語 L_1, L_2 は**余有限言語**である．

$$L_1 = \Sigma^* \setminus \{a, b, ab, aab\}$$

$$L_2 = \{w \in \Sigma^* \mid |w| \geq 3\}$$

Prefix Tree Automaton 有限言語は正規言語のサブクラスであることが知られており，DFA により認識可能である．そのような DFA を構成する方法の一つに，Prefix Tree Automaton (PTA) と呼ばれるアルゴリズムが存在する [10]．PTA ではまず，有限言語の prefix 集合 (例えば言語 $L = \{ab, baa\}$ に対して $\mathrm{Pref}(L) = \{\varepsilon, a, ab, b, ba, baa\}$ である) を状態とし，根が空文字列，葉が有限言語に含まれる文字列である木構造のような DFA を構成する．状態遷移関数 δ は $\delta(q_w, \sigma) = q_{w\sigma}$ で定義される．

アルゴリズム 1 に示すのは，DFA の遷移関数が全域関数となるように捨て状態 q_{trash} およびその周りの遷移 (8 行目) を加えたアルゴリズム PTA+ である．

また，ある DFA M の受理状態集合と非受理状態集合を入れ替えることで構成される DFA M' の認識する言語は，もとの言語の補集合であること，すなわち $L(M') = \Sigma^* \setminus L(M)$ であることが知られている．この性質を用いることで，任意の余有限言語に対してそれを認識する DFA も同様に構成することが可能である．

PTA+ の存在により，有限言語や余有限言語は正規言語の部分集合であることが確認できる．[1)]

3.2 p 精度浮動小数点数

本節では今後の全ての数値計算に用いられる p-精度浮動小数点数を定義する．

1) 実際には，これら (特に余有限言語) は正規言語クラスの**真部分集合**である．それは $\Sigma = \{a, b\}$ 上の言語 $L = a^*$ を考えれば明らかである．

Algorithm 1 Prefix Tree Automaton+

Input: Finite language L : set[string],
Alphabet Σ : set[char]
Output: DFA M such that $L(M) = L$

```
1: Q : set[state] ← {q_w | w ∈ Pref(L)} ∪ {q_trash}
2: Q_F : set[state] ← {q_w | w ∈ L}
3: δ : Callable[[state, char], state] ← {}
4: for all (q_w, σ) : tuple[state, char] ∈ Q × Σ do
5:     if q_wσ ∈ Q then
6:         δ(q_w, σ) ← q_wσ
7:     else
8:         δ(q_w, σ) ← q_trash
9:     end if
10: end for
11: M ← (Q, Σ, δ, q_ε, Q_F)
```

定義 4 (p-精度浮動小数点数 [11])**.** p-精度浮動小数点数の集合 $\mathbb{D}_p$ は p-bit の数の集まりであり，$\mathbb{D}_p = \{0, 1\}^p$ で表現される．なお，特殊な値として $+\mathrm{inf}, -\mathrm{inf}, \mathrm{nan} \in \mathbb{D}_p$ が存在する．また $\mathbb{D}_p$ は自然にベクトル ${\mathbb{D}_p}^*$ に拡張される．

定義 5 (p-精度浮動小数点演算 [11])**.** 関数 $f : {\mathbb{D}_p}^m \to {\mathbb{D}_p}^n : x_1, \ldots, x_m \mapsto y_1, \ldots, y_n$ が p-精度浮動小数点上の関数であるとは，f が p-空間チューリングマシンによって計算可能である場合である．

ここで $+\mathrm{inf}, -\mathrm{inf}, \mathrm{nan}$ に関する演算を含めた基本的な演算は IEEE754 形式 [12] に従う．

p の値が入力系列長 n を受け取り，精度を返す関数であり，定数関数 ($p(n) \in \mathcal{O}(1)$) であるときは定数精度，p が対数関数 ($p(n) \in \mathcal{O}(\log n)$) であるときは対数精度と呼ぶことにする．また，定数関数であるとき，定数 $p \in \mathbb{N}$ と同一視する．

3.3 Transformer Decoder

本節では Transformer decoder を関数として定式化する．

定義 6 (Transformer decoder)**.** 有限のアルファベットを Σ, 特殊トークン集合を $\mathbb{V}$ としたとき, Transformer decoder はパラメータ付き関数

$$\mathrm{TDec}_{p^t}(;\theta) : \Sigma^* \to (\Sigma \cup \mathbb{V})^* \tag{1}$$

である．関数を単に $\mathrm{TDec}(w)$ と書くこともある．

ここで下付き添字 p は全ての数値計算が p-精度浮動小数点代数上で行われることを意味し，θ は p-精度浮動小数点数のパラメータ集合である．また，出力系列の長さは上付き添字の時間関数 $t : \mathbb{N} \to \mathbb{N}$ によって制限され，入力系列 $w \in \Sigma^*$ に対して，$|\mathrm{TDec}(w)| = t(|w|)$ である．

なお，入力から出力までの計算の流れは基本

的に GPT モデル [13, 14] に従うが，本研究では**位置エンコーディングは不採用**であり，**2 層の Transformer Block**，**Single-head**，**pre-norm**，Attention 機構内の **softmax**，**Greedy Search**，**Causal masking** を採用する．

定義 6 は系列変換という Transformer decoder モデルの使用に基づく．定義域および終域について，系列変換モデルは一般に文脈を入力として受け取り，$\langle \mathrm{eos} \rangle \in \mathbb{V}$ 等の特殊トークンを含む系列へ変換する．

更に定義 6 は**浮動小数点数の精度**および**デコード回数上限**の制約を反映している．言語モデルは計算機上に実装されており，計算の精度は一般に fp32 や bf16 といった定数精度の浮動小数点代数が用いられ，先行研究 [3] のように入力系列長に応じた精度の増減は原則として行われない．また出力系列長は，入力系列長 n に対して何らかの時間関数 $t : \mathbb{N} \to \mathbb{N}$ により制限を行う．例えば定数 c に対して，時間関数が $t(n) = n^2$ の場合は多項式回のデコードを許容されるが，$t(n) = c$ の場合は入力の長さに関わらず定数回のデコードに制限される．

3.4 Transformer Decoder が認識する言語

Transformer decoder の関数としての定義 6 を踏まえ，同モデルを言語認識装置として定義する．

定義 7 (Transformer decoder が認識する言語)**.** ある停止状態トークン集合 $F \subseteq \mathbb{V}$ について，Transformer decoder が文字列 $w \in \Sigma^*$ を受理するのは，$\langle \mathrm{sep} \rangle \in \mathbb{V}$ を用いた出力系列 $\mathrm{TDec}_p{}^t(w\langle \mathrm{sep} \rangle; \theta) \in (\Sigma \cup \mathbb{V})^*$ 中に $v \in F$ を満たすトークン v が存在する場合，またその場合に限る．

終了状態トークン集合 F に対して Transformer decoder が認識する言語 $L(\mathrm{TDec}_p{}^t(;\theta), F)$ を，受理する文字列の集合で定義する．

本定義においては，デコード系列を明示的に区別するために入力系列の直後に $\langle \mathrm{sep} \rangle$ を挿入することに注意したい．

例 8. 時間関数が定数関数 $t(n) = 4$，停止状態トークン集合 $F = \{\langle \mathrm{eos} \rangle\}$ であり，入力系列 $aabb, aa$ に対する TDec の出力系列が

$$\mathrm{TDec}(aabb\langle \mathrm{sep} \rangle) = aba\langle \mathrm{eos} \rangle$$
$$\mathrm{TDec}(aa\langle \mathrm{sep} \rangle) = aaaa$$

であるとき，Transformer decoder は $aabb$ を受理し aa は拒否する．

4 結果

本節では，3 節の定義をもとに Trasnformer decoder が認識する言語が有限言語および余有限言語と一致することを示す．

まず自然な仮定を導入する．

仮定 9 (infinity-free パラメータ)**.** すべての Attention 層における query, key 積は任意の入力に対して常に負の無限大より大きな値を取る．すなわち下式が成立する．

$$\forall y, y' \in \mathbb{D}_p{}^d . Q(y)K(y')^\mathsf{T} > -\mathrm{inf} \tag{2}$$

ここで d はモデル次元，$Q, K : \mathbb{D}_p{}^d \to \mathbb{D}_p{}^d$ はそれぞれ query, key 変換である．

式 2 の真偽は Transformer のパラメータのみに依存し，本研究では成立するようなパラメータのみを考慮する．この仮定は一般的な学習済み Transformer モデルにおいて基本的に成立する．(付録 A を参照)

定理 10 (Transformer decoder が認識する言語の有限・余有限性)**.** 式 2 が成立することを仮定する．このとき，有限言語全体の集合と余有限言語全体の集合の和と Transformer decoder が認識する言語の集合が一致する．すなわち，以下の 2 つの主張が成立する．

1. 任意の $p \in \mathbb{N}, t(n) \in \Omega(n), \theta, F \subseteq \mathbb{V}$ について，ある有限言語または余有限言語 L_f が存在して $L(\mathrm{TDec}, F) \subseteq L_f$
2. 任意の有限言語または余有限言語 L'_f について，ある $p' \in \mathbb{N}, t'(n) \in \Omega(n), \theta', F' \subseteq \mathbb{V}$ が存在して $L'_f \subseteq L(\mathrm{TDec}, F')$

定理 10 は本研究における主要な結果であり，いかなるパラメータや n 回以上の任意回数のデコード，停止状態集合についても，仮定 9 が成立し定数精度 p であるならば，Transformer decoder が認識する言語クラスがちょうど有限言語または余有限言語の言語クラスと一致することを述べている．

証明　定理 10 の 2 つの主張について，それぞれ 4.1 節と 4.2 節で証明を行う．

4.1 $L(\mathrm{TDec}, F) \subseteq L_\mathrm{f}$ 証明

以下の補題を考える．

補題 11. ある長さ $l \in \mathbb{N}$ が存在して，$|w|, |w'| \geq l$ であるような任意の $w, w' \in \Sigma^*$ について $\mathrm{TDec}(w) = \mathrm{TDec}(w')$ が成立する．

証明　付録 B.1 を参照．　□

補題より l 以上の長さを持つ入力に対して，TDec は必ず同じ値を返す． Transformer が $|w| \geq l$ である文字列を受理する場合には 余有限言語を認識し，拒否する場合には有限言語を認識する．

4.2 $L'_f \subseteq L(\mathrm{TDec}, F')$ 証明

証明は先行研究 [8, 3] と同様に，DFA の動作の模倣という素朴な方法による． Attention 機構による入力文字 w_t の読み取り (補題 12) および FFN による遷移関数 $\delta : (q_{t-1}, w_t) \mapsto q_t$ の模倣 (補題 13) が可能であることを示し，帰納的に DFA の動作が模倣可能であることで証明を行う．

4.2.1 入力文字の読み取り

補題 12 (2 層 Attention による入力文字の取得)**.** 入力系列を $w_1 w_2 \cdots w_n \langle \mathrm{sep} \rangle q_0 \cdots q_{t-1}$，埋め込み層を $\mathrm{emb} : \Sigma \to {\mathbb{D}_p}^d$ とする．

このとき，q_{t-1} に対する第 2 層目 Attention 層出力 $\mathbf{h}^{(2)}_{n+t+1}$ および，任意の $\varepsilon > 0$ $(\in \mathbb{D}_p)$ について，

$$\left\| \mathbf{h}^{(2)}_{n+t+1} - \mathrm{emb}(w_t) \right\| < \varepsilon \tag{3}$$

を満たすパラメータが存在する．

本補題は 2 層の Attention 層を通じて，DFA が状態 q_{t-1} にて読み取る文字 w_t の埋め込みベクトルを を近似的に取り出せることを表す．

証明 詳細は付録 B.2 を参照. □

4.2.2 遷移関数の模倣およびトークン出力

補題 13 (遷移関数の模倣およびトークン出力)**.** q_{t-1} に対する第 2 層の Attention 出力 $h^{(2)}_{n+t+1}$ と， residual connection 経由で保持される前状態 q_{t-1} の埋め込み $\mathrm{emb}(q_{t-1})$ および，任意の $\varepsilon > 0$ $(\in \mathbb{D}_p)$ について，

$$\left\| \mathrm{FFN}(\mathbf{h}^{(2)}_{n+t+1}, \mathrm{emb}(q_{t-1})) - \mathrm{emb}(q_t) \right\| < \varepsilon \tag{4}$$

を満たすパラメータが存在する． さらに，出力層により q_t トークンがデコードされる．

本補題は単層の FFN により DFA の遷移関数を近似的に模倣でき，出力層においてそのノイズを除去できることを表す．

証明 詳細は付録 B.3 を参照. □

以上の補題により，$L_f \subseteq L(\mathrm{TDec}, F)$ 方向も示され，求める定理 10 を示すことができた．

5 議論

結果の拡張の見通し 表 1 (1) は，補題 11 および補題 12 の softmax 関数を hardmax 関数に置き換えることで示すことができる． 後者の補題 12 より強い補題が多くの先行研究 [8, 3] により示されており，前者の補題 11 に関しても l を適切に設計することで容易に示すことができると考えられる．

また表 1 (4) も同様にして示すことができる． この場合は補題 12 および 補題 13 で行った議論を，対数精度の場合 [3] に適応させることで示すことができると考えるが，証明は自明ではないため今後の研究に期待したい．

本節で述べた予想が正しい場合，表現力の議論において softmax 関数と hardmax 関数に本質的な影響力がないことが，逆に精度を対数から定数へ制限することで，大幅に表現力が低下することが示される． そして表現力の低下は，$t \in \mathcal{O}(n^c)$ のときに P である [3] ことに注意すると，デコード回数のオーダーを増やした場合に特に顕著となる．

言語モデリングとの乖離 本研究では Transformer のトークン集合をアルファベットと特殊トークン集合の和 $\Sigma \cup \mathbb{V}$ とし，また Transformer decoder を言語認識装置として所属性問題，すなわちある言語に所属するかという問題を念頭に置いていた． しかし，実際のトークン集合は DFA の状態集合 Q を含まない少数のトークンで構成されており，また Transformer decoder の本来の用途で文字列に確率を与える言語モデリングとしての使用からかけ離れたものである．

6 結論

本研究では Attention 機構内に softmax 関数を採用し，p-精度浮動小数点数に基づき実装された Transformer decoder を言語認識装置として定義した (3.4 節)． そのときに認識する言語クラスが有限言語および余有限言語 (3.1 節) クラスと一致することを示した (4 節, 定理 10)． さらに 5 節では，本研究における証明手法の汎用性や限界について論じた．

以上により Transformer decoder が取り扱う言語クラスの性質を明確にし，今後の発展的研究に向けた基盤を提供できたと考える．

謝辞

本研究は AMED の課題番号 24wm0625405h0001 の助成および JSPS 科研費の課題番号 JP24K16077 の助成を受けたものです.

参考文献

[1] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Ł 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] Lena Strobl, William Merrill, Gail Weiss, David Chiang, and Dana Angluin. What formal languages can transformers express? a survey. **Transactions of the Association for Computational Linguistics**, Vol. 12, pp. 543–561, 2024.

[3] William Merrill and Ashish Sabharwal. The expressive power of transformers with chain of thought. In **The Twelfth International Conference on Learning Representations**, 2024.

[4] Jason Wei, Xuezhi Wang, Dale Schuurmans, Maarten Bosma, Brian Ichter, Fei Xia, Ed H. Chi, Quoc V. Le, and Denny Zhou. Chain-of-thought prompting elicits reasoning in large language models. In **Proceedings of the 36th International Conference on Neural Information Processing Systems**, NIPS '22, Red Hook, NY, USA, 2024. Curran Associates Inc.

[5] George V. Cybenko. Approximation by superpositions of a sigmoidal function. **Mathematics of Control, Signals and Systems**, Vol. 2, pp. 303–314, 1989.

[6] Chulhee Yun, Srinadh Bhojanapalli, Ankit Singh Rawat, Sashank Reddi, and Sanjiv Kumar. Are transformers universal approximators of sequence-to-sequence functions? In **International Conference on Learning Representations**, 2020.

[7] David Chiang, Jon Rawski, Lena Strobl, and Andy Yang. Esslli 2024, （2025-01 閲覧）. https://sleynas.com/esslli-2024-summer-school-course.

[8] Jorge Pérez, Pablo Barceló, and Javier Marinkovic. Attention is turing-complete. **Journal of Machine Learning Research**, Vol. 22, No. 75, pp. 1–35, 2021.

[9] Zhiyuan Li, Hong Liu, Denny Zhou, and Tengyu Ma. Chain of thought empowers transformers to solve inherently serial problems. In **The Twelfth International Conference on Learning Representations**, 2024.

[10] J. Oncina and P. García. **INFERRING REGULAR LANGUAGES IN POLYNOMIAL UPDATED TIME**, pp. 49–61.

[11] William Merrill and Ashish Sabharwal. The parallelism tradeoff: Limitations of log-precision transformers. **Transactions of the Association for Computational Linguistics**, Vol. 11, pp. 531–545, 2023.

[12] Ieee standard for floating-point arithmetic. **IEEE Std 754-2019 (Revision of IEEE 754-2008)**, pp. 1–84, 2019.

[13] Alec Radford and Karthik Narasimhan. Improving language understanding by generative pre-training. 2018.

[14] Tom B. Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, Sandhini Agarwal, Ariel Herbert-Voss, Gretchen Krueger, Tom Henighan, Rewon Child, Aditya Ramesh, Daniel M. Ziegler, Jeffrey Wu, Clemens Winter, Christopher Hesse, Mark Chen, Eric Sigler, Mateusz Litwin, Scott Gray, Benjamin Chess, Jack Clark, Christopher Berner, Sam McCandlish, Alec Radford, Ilya Sutskever, and Dario Amodei. Language models are few-shot learners. In **Proceedings of the 34th International Conference on Neural Information Processing Systems**, NIPS '20, Red Hook, NY, USA, 2020. Curran Associates Inc.

[15] Alec Radford, Jeff Wu, Rewon Child, David Luan, Dario Amodei, and Ilya Sutskever. Language models are unsupervised multitask learners. 2019.

[16] Ben Wang and Aran Komatsuzaki. GPT-J-6B: A 6 Billion Parameter Autoregressive Language Model. https://github.com/kingoflolz/mesh-transformer-jax, May 2021.

[17] Amirhossein Kazemnejad, Inkit Padhi, Karthikeyan Natesan Ramamurthy, Payel Das, and Siva Reddy. The impact of positional encoding on length generalization in transformers. In **Proceedings of the 37th International Conference on Neural Information Processing Systems**, NIPS '23, Red Hook, NY, USA, 2024. Curran Associates Inc.

[18] Sejun Park, Chulhee Yun, Jaeho Lee, and Jinwoo Shin. Minimum width for universal approximation. In **International Conference on Learning Representations**, 2021.

A　仮定 9 の妥当性

仮定 9 内の式 2 について，今回のように pre-layer normalization を採用する場合 $\|y\|, \|y'\| \sim 1$ であるため，$|Q(y)K(y')^\top| \le (\|W_Q\| + \|b_Q\|)(\|W_K\| + \|b_K\|) < \mathrm{Inf}$ と変形できる．初期パラメータの選択や，学習過程における正規化・勾配クリッピング等の技術によりこの不等号は基本的に成立する．

実際に事前学習済みモデル [15, 16] の各 Attention 層のノルム和積 $\gamma \overset{\mathrm{def}}{=} (\|W_Q\| + \|b_Q\|)(\|W_K\| + \|b_K\|)$ を計算してみると，表 2 のように最大値 ($\gamma_{\max}$) もオーバーフローしない小さな値に収まっていた．

表 2　Decoder モデルの Attention 層のノルム和積の最大値・最小値

モデル名	$\gamma_{\min}$	$\gamma_{\max}$
gpt-2 small	420.94 (layer-12)	1974.25 (layer-5)
gpt-2 large	182.80 (layer-21)	283.92 (layer-15)
gpt-j-6b	3630.00 (layer-24)	6318.29 (layer-1)

B　証明

B.1　補題 11 の証明

まず l を以下で定義する．

$$l = \min\left\{ l \;\middle|\; \sum^{l} Q(\alpha)K(\beta)^\top = +\mathrm{inf} \right\} \tag{5}$$

ここで埋め込み関数 $\mathrm{emb} : \Sigma \to \mathbb{D}_p{}^d$ を用いて，

$$\alpha = \mathrm{emb}(\langle \mathrm{sep} \rangle), \beta = \mathrm{emb}\left(\underset{\tau \in \Sigma}{\mathrm{argmin}} \left(Q(\alpha)K(\mathrm{emb}(\tau))^\top \right) \right) \tag{6}$$

であり，α ならびに β は定数である．l は l 以上の長さの文字列について，必ず query, key 積の和が無限大の大きさになることを保証する値であり，仮定 9 よりその存在が保証される．

デコード開始のための $\langle \mathrm{sep} \rangle$ トークンに対する Attention 層の出力は，

$$\mathrm{Attn}(Q_{\langle \mathrm{sep} \rangle}, K, V) = \frac{\exp Q_{\langle \mathrm{sep} \rangle} K^\top}{\sum \exp\left(Q_{\langle \mathrm{sep} \rangle} K^\top \right)} V \tag{7}$$

であるが，softmax 関数の分母に注目すると，

$$\sum_{j}^{|w|+1} \exp\left(QK_j \right) \ge \sum^{l} \exp\left(Q(\alpha)K(\beta) \right) = +\mathrm{inf} \tag{8}$$

と評価されるため，$\mathrm{Attn}(Q_{\langle \mathrm{sep} \rangle}, K, V) = \mathbf{0}$ となる．

また Transformer では文脈を引数に取る関数が attention 機構以外に存在しないため，帰納的に $\mathrm{TDec}(w) = \mathrm{TDec}(w')$ が成立する．[2] よって補題は示された．

B.2　補題 12 の証明

(1) 第 1 層による位置分化 [17]　第 1 層では各入力トークン w_i を，causal masking のもとで適切に設計された transformer block Block_1 に適用する．このとき十分な精度があれば，

$$\mathbf{z}_i^{(1)} = \mathrm{Block}_1\left(\mathrm{emb}(w_i)\right) \tag{9}$$

が，位置 i の違いに応じて異なるベクトルになるよう調整することができる．すなわち同じ文字 $w_i = w_j$ であっても $i \ne j$ なら $\mathbf{z}_i^{(1)} \ne \mathbf{z}_j^{(1)}$．

(2) 第 2 層による一点的注意　次に，時刻 t の query ベクトルを $Q_t \overset{\mathrm{def}}{=} \mathbf{z}_{q_{t-1}}^{(1)} W_2^Q$ とし，key を $K_i \overset{\mathrm{def}}{=} \mathbf{z}_i^{(1)} W_2^K$ とする．ここで $i \ne t$ の場合には $Q_t K_i^\top < Q_t K_t^\top$ となるよう，行列 W_2^Q, W_2^K を設定できる．温度 τ を十分大きくとった attention スコア

$$\alpha_i = \frac{\exp(\tau \cdot Q_t K_i^\top)}{\sum_j \exp(\tau \cdot Q_t K_j^\top)} \tag{10}$$

は，$i = t$ に近似的に 1 の重みを与える．

(3) 出力ベクトルの支配性　第 2 層目 attention 層の出力 $\mathbf{h}_{n+t+1}^{(2)}$ は，attention スコアによる重み付き value 和 $\sum_i \alpha_i \mathbf{v}_i$ であり，$\alpha_t \approx 1$ ならば $\|\mathbf{h}_{n+t+1}^{(2)} - \mathrm{emb}(w_t)\| < \varepsilon$ を満たせる．有限長 n ならパラメータのスケーリングを調整することで誤差を任意に小さくできる．

以上により，q_{t-1} デコード時における第 2 層目 attention 層により，w_t の埋め込みをほぼ一点的に抽出可能であり，補題の主張が成り立つ．

B.3　補題 13 の証明

(1) FFN による遷移関数の模倣．　DFA において，遷移関数 $\delta : Q \times \Sigma \to Q$ は定義域・値域が有限集合である．よって，任意の $(\mathrm{emb}(q_{t-1}), \mathrm{emb}(w_t))$ の組に対して $\mathrm{emb}(q_t)$ を返す写像を，FFN (多層パーセプトロン) を通じて任意の精度で近似できる [18]．具体的には，埋め込みベクトルを結合した $\left[\mathbf{h}_{n+t+1}^{(2)}, \mathrm{emb}(q_{t-1})\right]$ を入力として，線形変換と ReLU 等の非線形を組み合わせれば，$\mathrm{emb}(q_t)$ に近いベクトル $\mathbf{z}_{n+t+1}^{(2)}$ を出力可能になる．

(2) 出力層におけるトークンの決定．　FFN により $\|\mathbf{z}_{n+t+1}^{(2)} - \mathrm{emb}(q_t)\| < \varepsilon$ である $\mathbf{z}_{n+t+1}^{(2)}$ が得られたとする．ここで，出力層として Layer Normalization と線形変換を施し，$\Sigma \cup \mathbb{V}$ ($\mathbb{V} \supseteq Q$) 上で argmax を取ると，トークン q_t を一意に確定できる．

2)　この結果を拡張すると，$\langle \mathrm{sep} \rangle$ トークンのように入力文字列直後に特殊トークンを挟まないようなモデルの表現力の上限は末尾による区別が可能な言語まで拡大されることがわかる．例) $L = \{w \in \Sigma^* \mid$ 末尾がaであるような言語$\}$