部分空間法に着想を得たTransformer のアテンションヘッドにおける特徴抽出

前田晃弘 [1]　鳥居拓馬 [2]　日髙昇平 [1,3]　大関洋平 [4]
[1] 北陸先端科学技術大学院大学　[2] 東京電機大学　[3] ロンドン大学シティ校　[4] 東京大学
{akihiro.maeda, shhidaka}@jaist.ac.jp
tak.torii@mail.dendai.ac.jp,　oseki@g.ecc.u-tokyo.ac.jp

概要

Transformer をベースとした言語モデルは，幅広い自然言語処理タスクにおいて高い性能を示している．本研究は，文における単語分散表現の合成メカニズムを解明するため，Transformer のアテンションヘッドに注目し，その内部計算を部分空間への射影と捉え，ノルムの変化率により各ヘッドで捕捉される言語的な特徴を同定する新たな手法を提案する．事前学習済み BERT を用いた実証実験では，各アテンションヘッドが異なる特徴を抽出している可能性が示唆される．

1　はじめに

Transformer [1] をベースとした大規模言語モデル (LLM) が人間並の意味理解や意味生成を実現するなど高い性能を示している．LLM は単語ベクトルを入力とした演算を内部で行い質問応答などのタスクを実行するが，モデルが複雑であることから，その内部計算の解釈は困難である．Transformer ベースの言語モデルに関して，その内部解明を試みる研究は BERTology [2] と呼称されるほど盛んであり，分類器を用いて符号化されている言語的情報を調べるなど多くの研究手法が提案されている [3]．特に，アテンションと呼ばれる計算過程（自己注意機構）の分析から，Transformer が文の句構造など統語的関係を捕捉していると考えられる [4]．

本研究では，Transformer ベースの言語モデルにおける単語分散表現の合成メカニズムを明らかにすることを目的として，アテンション内部における計算過程の解釈に取り組む．アテンションヘッドと呼ばれるサブユニットでは，入力ベクトルを低次元ベクトルへ変換した上で，ベクトル間の演算が行われる．この線型写像は特徴抽出を行う部分空間への射影と見做すことができる．部分空間法と呼ばれるパターン認識の手法 [5] においてノルムを最大化する部分空間を用いて特徴ベクトルのクラス分類を行うことに着想を得て，射影によるノルムの変化率に着目した新たな指標を定義する．その上で，この指標を用いてヘッドに対応する部分空間が抽出している言語的特徴を同定する手法を提案する．Transformer ベースの言語モデルの一つである BERT[6] の学習済みモデルを用いた実験において，各アテンションヘッドが異なる言語的特徴を抽出している可能性があることを示す．

2　分析のためのアプローチ

2.1　Transformer の概要

Transformer [1] は，自己注意機構を特徴とする深層学習モデルである．各レイヤーは，アテンションとフィードフォワードと呼ばれる二つのサブレイヤーから構成される．入力文の各単語に対応する単語埋め込みとトークンの位置を表す位置ベクトルの和を状態ベクトルとして，レイヤーでの演算処理が行われる．

アテンションは，文中のトークン間の関連性を，対応するベクトル間の内積により評価した上で，その関連単語のベクトルを元単語の状態ベクトルに重み付け加算する．そのウェイトは，動詞-目的語の依存関係など文の統語構造を反映していることが知られており [4, 7]，アテンションは文における意味合成機能を担っていると考えられる．

一方，フィードフォワードは，事前学習された内部パラメータに基づいて算出されたベクトルを加算し，状態ベクトルを更新する．内部パラメータには学習時に獲得された世界知識や言語知識が記憶されており [8]，主題の肥沃化 (subject enrichment) が行わ

れていると考えられる [9].

状態ベクトルが各レイヤーにおいて算出されたベクトルにより加算・更新されていくプロセスを residual stream と捉え [10]，加算されるベクトルを言語的に解釈する分析手法が提案されている [11].

2.2 アテンションにおける計算過程

入力文のトークン数[1)]を n，各トークンに対応する状態ベクトルを $x \in R^d$ とする．この時，各レイヤーに入力処理されるのは，状態ベクトル（行ベクトル）を垂直にスタックした行列 $X \in R^{n\times d}$ となる．$l \in \{1,\ldots,L\}$ 番目のレイヤーに入力される状態ベクトルを集めた行列を X_l とする．

各レイヤーの $h \in \{1,\ldots,H\}$ 番目のアテンションヘッドでは，状態ベクトルを query, key, value の 3 つの d' 次元ベクトルへ線型写像する（式 1）.

$$Q_l^h = X_l W_{q_l^h}, \quad K_l^h = X_l W_{k_l^h}, \quad V_l^h = X_l W_{v_l^h} \tag{1}$$

$Q_l^h, K_l^h, V_l^h \in \mathbb{R}^{n\times d'}$ は文中のトークンに対応する 3 つのベクトルをスタックした行列であり，それらを得る線型写像を表現する行列 $W_{q_l^h}, W_{k_l^h}, W_{v_l^h} \in \mathbb{R}^{d\times d'}$ はモデル訓練時に学習される．$d' = \frac{d}{H}$（ただし，H はヘッド数）である．次に，query と key の内積を計算し，それぞれのトークンが文中の他の単語に対して向けるべきアテンションのウェイトを表す行列 A_l^h が式 (2) により与えられる.

$$A_l^h = \mathrm{softmax}\left(\frac{Q_l^h K_k^{hT}}{\sqrt{d'}}\right) \tag{2}$$

最後に，value ベクトルの重み付き総和を $V_l^h A_l^h$ により求め，これを行列 $W_{o_l^h} \in \mathbb{R}^{d'\times d}$ により d 次元へ戻すことでアテンションの出力 Y_l^h を得る（式 3）.

$$Y_l^h = V_l^h A_l^h W_{o_l^h} \tag{3}$$

複数の並行的なアテンションヘッドは，文中間で一つのトークンが複数の統語的関係に関与する状況を捕捉しているとされる [12].

2.3 部分空間法からの着想

パターン認識における部分空間法は，特徴ベクトルが高次元空間の中の非常に次元の小さい部分空間に偏在することに着目して，部分空間を利用してクラス分類を行う手法である [5].

BERT の base モデルの場合[2)]，query, key, value を求める行列計算は，768 次元のベクトル空間から 64 次元への線型写像であり，部分空間への射影と考えることができる．図 1 は，3 つのベクトル

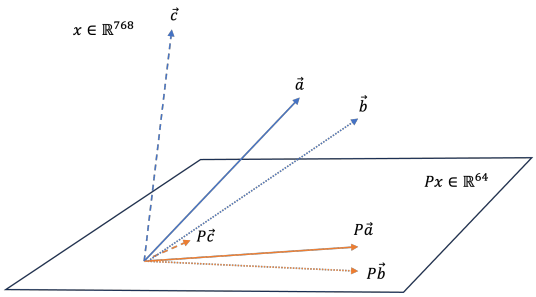

図 1 各ヘッドの部分空間への射影

$\vec{a}, \vec{b}, \vec{c} \in \mathbb{R}^{768}$ が行列 $P \in \mathbb{R}^{64\times 768}$ により写像され，部分空間において 3 つのベクトル $P\vec{a}, P\vec{b}, P\vec{c} \in \mathbb{R}^{64}$ が得られる様を示す．ベクトル $\vec{a}, \vec{b}$ は，射影後の部分空間においても十分なノルムを持つので，共通の特徴を有していると見られる一方，そうでない $\vec{c}$ のノルムは減退する．部分空間法では，各クラスごとにそのクラスを表現する低次元の部分空間を用意し，未知サンプルを射影した際に最大ノルムを示す部分空間に対応するクラスへ分類する.

これを逆に考えると，行列 $W_{q_l^h}, W_{k_l^h}, W_{v_l^h}$ による線型写像は，それぞれ異なる特徴を抽出する部分空間のクラスに対応していると考えられる．従って，ノルムの伸縮を調べることにより，それぞれのヘッドが抽出している特徴を同定できると考えられる.

3 提案手法

3.1 部分空間のノルム伸縮率の定義

ベクトル $x \in \mathbb{R}^d$ を行列 $P \in \mathbb{R}^{d'\times d}$ により d' 次元ベクトル Px へ射影するとき，射影前後でそのノルムを比較するための指標として，ノルムの伸縮率 ψ を式 (4) により定義する.

$$\psi := \frac{\|Px\|}{\sqrt{d'}} \Big/ \frac{\|x\|}{\sqrt{d}} \tag{4}$$

定義域と値域の次元数の違いを反映して，ノルムを次元数の二乗根で正規化している．例えば，$(1,1,1)^T \mapsto (1,1)^T$ の場合，正規化後のノルムは $\sqrt{3}/\sqrt{3} = 1$ と $\sqrt{2}/\sqrt{2} = 1$ となり，ノルム伸縮率 $\psi = 1.0$ となる．正規分布に従う値を持つ高次元ベクトルの場合，次元数 d が大きくなるとノルムは $\sqrt{d}$ に近づく（球面集中現象）ので，次元数の二乗根によりノルムは正規化される．ψ が大きいベクトルほど当該部分空間が抽出する特徴を有し，ψ が小さいベクトルほど特徴を有しないと解される.

1) 文頭やパディングを表すタグを含む.
2) ハイパーパラメータ $L = 12, H = 12, d = 768$

ψ の二乗は正規化したレイリー商である．

$$\frac{\|Px\|^2}{\|x\|^2} = \frac{(Px)^T Px}{x^T x} = \frac{x^T Mx}{x^T x} =: R(M,x) \qquad (5)$$

但し，$M = P^T P$ はエルミート（対称）行列である．レイリー商 $R(M,x)$ は，その最大（小）値が行列 M の最大（小）固有値に等しいという性質を持つ [13]．ψ はレイリー商の定数（$\frac{d}{d'}$）倍の二乗根であるので，ノルム伸縮率は行列 M の最大固有値に対応する固有ベクトルに沿うような分散表現を持つ単語を抽出していると言える．

アテンションの各ヘッドでは，query, key, value それぞれの部分空間への射影が行われているため，$P = W_{q_l^h}, W_{k_l^h}, W_{v_l^h}$ として，それぞれごとにノルム伸縮率 ψ_q, ψ_k, ψ_v を計算する．[14] は，アテンションの出力ベクトルに対して，query と key の内積から得られるウェイトだけではなく，value のノルムの大きさもインパクトをもたらす因子であることを指摘している．すなわち，一見ウェイトが大きな単語間の組み合わせであっても，value のノルムが小さいためアテンションの出力全体に影響を与えないケースがある．これを踏まえて，本研究では query, key, value の 3 つの部分空間における統合的な特徴抽出を評価するため，3 つのノルム伸縮率の積 $\psi_{qkv} := \psi_q \cdot \psi_k \cdot \psi_v$ を用いる．なお，query のトークンと，key, value のトークンは一般に異なるので，ψ_{qkv} の値に関わらず，組み合わせ次第で出力にインパクトを与える可能性がある．同一ベクトルに対するノルム伸縮率のみに着目した本指標は，ヘッドが抽出する特徴を近似的に同定するためのものであることに留意する．

3.2 単語埋め込みの類型化

前節に定義したノルム伸縮率はトークン単位で計算されるが，各ヘッドが傾向的にどのような言語的特徴を抽出しているかを調べるために，トークンをグループ分けした上で，ヘッドで高い伸縮率を示すグループを特定することを試みる．BERT の入力に用いられる隠れ状態ベクトルには，WordPiece と呼ばれる，機械翻訳のためにあらかじめ学習された静的な単語埋め込み [15] が用いられている．従って，BERT が用いているトークン数 30,522 の語彙に対応した単語埋め込みに対して k-means 法 [16] によるクラスタリング（クラスタ数を 1000 と設定）を行う．クラスタリングの結果例を表 1 に示す．

クラスタリング結果に基づいて，それぞれのクラスタをさらに著者がマニュアルでタイプ分けを行なった（表 1 中の Type として 6 種類を表示）．結果例に示すように，クラスタは，類似するトークンのタイプや，Words タイプであれば品詞などの統語的範疇や意味的範疇の観点で類似度の高い単語を含んでいることが観察された．クラスタリング結果全体の概要を付録 A に示す．

表 1 単語埋め込みのクラスタ例（クラスタ数 1000）

Type	ID	Samples
Words	3	*anger, rage, fury, temper*
	251	*jeans, pants, shorts, trousers, slacks*
	483	*keep, keeps, kept, keeping*
	522	*of, it, to, for, with, by, at, from, up, out, ...*
	606	*daily, weekly, annually, monthly, yearly,...*
	647	*france, india, canada, germany, japan, ...*
Subwords	116	*##ist, ##ism, ##ists, ##istic*
	481	*##rate, ##rated,##rating, ##ration,...*
Symbols	419	*[SEP], !, ”, ’, (,), -, ., :,;...*
Numbers	679	*1790, 1789, 1776, 1775, 1791, 1780, ...*
Special	924	*[PAD], [unused0],[unused1],[unused2],...*
Tags	268	*[CLS], [MASK]*

ユークリッド距離の近いベクトルは写像における挙動も類似することから，クラスタのノルム伸縮率を見ることで，クラスタ共通の言語的特徴が部分空間でどのように抽出されているかを調べることが可能である．

モデルに入力する状態ベクトルは，単語埋め込みと位置ベクトルの和であるところ (2.1 節)，両ベクトル間の直交性を踏まえ（付録 B），本研究では単語埋め込み部分に対してノルム伸縮率を計算する．

4 実験

4.1 データ

事前学習済み言語モデルとして，Hugging face が提供する BERT_base_uncased[17] を用いる．同モデルのハイパーパラメータは，$L = 12, H = 12, d = 768, d' = 64$ である．全 144 個 $(L \times H)$ のヘッドごとに 3 つの射影に対応する行列の学習済みパラメータを抽出して用い，バイアス項は用いない．また前節 3.2 の通り，単語埋め込み (30,522 トークン）も同モデルのものを用いた．

4.2 クラスタ別のノルム伸縮率

各ヘッドごとにトークンのノルム伸縮率 ψ_{qkv} を計算し，クラスタに含まれるトークンの平均値をクラスタのノルム伸縮率とした．図 2 に示す 12 個のグラフは各レイヤーに対応し，それぞれの折れ線は

12 のヘッドに対応する．いずれのヘッドにおいて

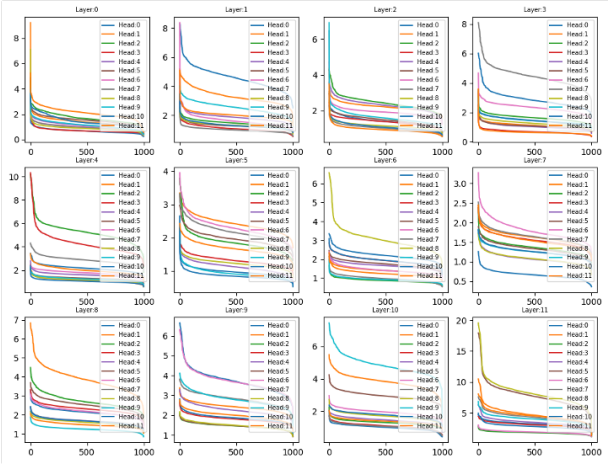

図 2　ヘッド別ノルム伸縮率（横軸クラスタ降順）

も，グラフは上位少数のクラスタにおいて大きな値を示したのち，他のクラスタ領域では平坦である．アテンションウェイトの算出に用いられる Softmax 関数の性質より，各ヘッドは少数の選別されたクラスタを抽出していることが示唆される．

4.3　結果と分析

表 2 に，ヘッドにおけるノルム伸縮率上位のクラスタに属するトークンの例を示す．例えば，L3-H2

表 2　ヘッドが抽出する特徴を持つクラスター例

Head	ID	Tokens	ψ_{kqv}
L3-H2	357	*2010,2011,2012,2013,2014, ...*	3.53
	332	*january, february, april, may, ...*	2.74
	347	*1987,1988,1989, 1990, ...*	2.72
	199	*march*	2.66
	398	*/*	2.53
L3-H11	268	*[CLS],[MASK]*	2.17
	675	*a,the,it,that, he,his,she,her, ...*	2.14
	419	*[SEP], !, ",',(,),...*	2.14
	523	*i,you, we, me, us, him*	2.12
	176	*and, as, but, or, is, are, ...*	2.11
L4-H6	688	*teach, teaches, taught, teaching*	2.41
	833	*announce, announces, announced, ...*	2.40
	398	*/*	2.37
	528	*do, does, did, done, going, saying, ...*	2.34
	80	*welcome, welcomed, welcoming*	2.33

（レイヤー 3 におけるヘッド番号 2 の略記．以下同じ．）のヘッドにおいて大きなノルム伸縮率を示したクラスタは，西暦を表すと見られる Numbers タイプのクラスタ（ID357, ID347）および暦月を表す Words タイプのクラスタ（ID332, ID199）であり，「年月」という意味的特徴が抽出されていると考えられる．同様に，L3-H11 は，句読点記号やタグのクラスタ (ID268, ID419) と冠詞・代名詞・接続詞などの機能語のクラスタ (ID675, 523, 176) を抽出しており，文構造を捕捉していると見られる．また，L4-H6 は活用形を含む動詞単語のクラスタを抽出している．

図 3 は，全ヘッドにおけるノルム伸縮率上位 3 つのクラスタのトークンタイプを図示しており，レイヤー別の傾向が明らかである．低層（L0-L3）では，

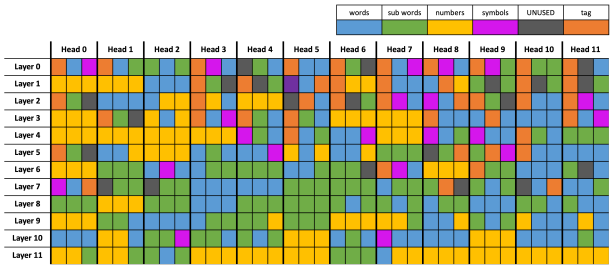

図 3　ψ_{qkv} 上位 3 タイプ（縦軸レイヤー；横軸ヘッド）

Symbols, Tags, Special タイプに属するクラスタが多く出現している．また，Words タイプでも機能語のクラスタが多く出現している．低層のヘッドでは文の構造把握を行なっていることが推察される．中層（L4-L7）では，Words と Subwords タイプのクラスタが多く見られるようになり，形態素から意味を把握する，あるいは単語を組み合わせ句の意味を合成するなどの処理が行われていることが示唆される．高層 (L8-11）では，Words と Numbers タイプのクラスタが中心であり，一つのヘッドの中でも単一のタイプのみが観察されるようになる．なお，異なるタイプが混在するヘッドでも実質的に共通の特徴をもつクラスタを抽出している場合がある．

5　考察と結論

本研究の提案するノルム伸縮率を用いた分析手法により，Transformer のアテンションヘッドが解釈可能な言語的特徴を共有するクラスタを抽出していることが明らかになった．Transformer が文の意味理解や意味合成を実現していることを踏まえると，各ヘッドはそのために必要な言語的特徴を抽出していることが示唆される．

今後の研究として，単語ベクトルの意味合成メカニズムを解明する観点からは，Words タイプのクラスタに対応するヘッドにおいて，高いノルム伸縮率を示す単語の組み合わせにおいて実行されている状態ベクトルの更新をトークンレベルで分析することが考えられる．その際，位置ベクトルがアテンションにおいて相対位置を指示する [18] こと，また位置ベクトルと単語埋め込みは直交していること（付録 B）を踏まえた分析を行う必要がある．

謝辞

本研究は科研費基盤研究 B（一般）JP23H0369, JST さきがけ JPMJPR20C9, JPMJPR21C2, JST CREST JPMJCR23P4 の助成を受けて行われた.

参考文献

[1] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Lukasz Kaiser, and Illia Polosukhin. Attention is all you need. **Advances in neural information processing systems**, Vol. 30, , 2017.

[2] Anna Rogers, Olga Kovaleva, and Anna Rumshisky. A primer in BERTology: What we know about how BERT works. **Transactions of the Association for Computational Linguistics**, Vol. 8, pp. 842–866, 2020.

[3] Yonatan Belinkov and James Glass. Analysis methods in neural language processing: A survey. **Transactions of the Association for Computational Linguistics**, Vol. 7, pp. 49–72, 2019.

[4] Kevin Clark, Urvashi Khandelwal, Omer Levy, and Christopher D. Manning. What does BERT look at? an analysis of BERT's attention. In Tal Linzen, Grzegorz Chrupa la, Yonatan Belinkov, and Dieuwke Hupkes, editors, **Proceedings of the 2019 ACL Workshop BlackboxNLP: Analyzing and Interpreting Neural Networks for NLP**, pp. 276–286, Florence, Italy, August 2019. Association for Computational Linguistics.

[5] 石井健一郎, 上田修功, 前田英作, 村瀬洋. わかりやすいパターン認識. オーム社, 2001.

[6] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. BERT: Pre-training of deep bidirectional transformers for language understanding. In Jill Burstein, Christy Doran, and Thamar Solorio, editors, **Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)**, pp. 4171–4186, Minneapolis, Minnesota, June 2019. Association for Computational Linguistics.

[7] John Hewitt and Christopher D. Manning. A structural probe for finding syntax in word representations. In Jill Burstein, Christy Doran, and Thamar Solorio, editors, **Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)**, pp. 4129–4138, Minneapolis, Minnesota, June 2019. Association for Computational Linguistics.

[8] Mor Geva, Avi Caciularu, Kevin 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**, pp. 30–45, Abu Dhabi, United Arab Emirates, December 2022. Association for Computational Linguistics.

[9] Mor Geva, Jasmijn Bastings, Katja Filippova, and Amir Globerson. Dissecting recall of factual associations in auto-regressive language models. In Houda Bouamor, Juan Pino, and Kalika Bali, editors, **Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing**, pp. 12216–12235, Singapore, December 2023. Association for Computational Linguistics.

[10] Nelson Elhage, Neel Nanda, Catherine Olsson, Tom Henighan, Nicholas Joseph, Ben Mann, Amanda Askell, Yuntao Bai, Anna Chen, Tom Conerly, Nova DasSarma, Dawn Drain, Deep Ganguli, Zac Hatfield-Dodds, Danny Hernandez, Andy Jones, Jackson Kernion, Liane Lovitt, Kamal Ndousse, Dario Amodei, Tom Brown, Jack Clark, Jared Kaplan, Sam McCandlish, and Chris Olah. A mathematical framework for transformer circuits. **Transformer Circuits Thread**, 2021. https://transformer-circuits.pub/2021/framework/index.html.

[11] Guy Dar, Mor Geva, Ankit Gupta, and Jonathan Berant. Analyzing transformers in embedding space. In Anna Rogers, Jordan Boyd-Graber, and Naoaki Okazaki, editors, **Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)**, pp. 16124–16170, Toronto, Canada, July 2023. Association for Computational Linguistics.

[12] Ashish Vaswani. Stanford CS224N: Transformers and self-attention, 2019. https://youtu.be/5vcj8kSwBCY?si=MWmbBwoBUekksSuR.

[13] Grégoire Allaire, Sidi Mahmoud Kaber, Karim Trabelsi, and Grégoire Allaire. **Numerical linear algebra**, Vol. 55. Springer, 2008.

[14] Goro Kobayashi, Tatsuki Kuribayashi, Sho Yokoi, and Kentaro Inui. Attnetion is not only a weight: Analyzing transformers with vector norms. In **Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing**, pp. 7057–7025, 2020.

[15] Yonghui Wu, Mike Schuster, Zhifeng Chen, Quoc V. Le, Mohammad Norouzi, Wolfgang Macherey, Maxim Krikun, Yuan Cao, Qin Gao, Klaus Macherey, Jeff Klingner, Apurva Shah, Melvin Johnson, Xiaobing Liu, Lukasz Kaiser, Stephan Gouws, Yoshikiyo Kato, Taku Kudo, Hideto Kazawa, Keith Stevens, George Kurian, Nishant Patil, Wei Wang, Cliff Young, Jason Smith, Jason Riesa, Alex Rudnick, Oriol Vinyals, Greg Corrado, Macduff Hughes, and Jeffrey Dean. Google's neural machine translation system: Bridging the gap between human and machine translation, 2016.

[16] scikit learn. sklearn.cluster.kmeans, 2024. https://scikit-learn.org/stable/modules/generated/sklearn.cluster.KMeans.html.

[17] Hugging Face. BERT base model (uncased), 2024. https://huggingface.co/bert-base-uncased.

[18] 山本悠士, 松崎拓也. 自己注意機構における注意の集中が相対位置に依存する仕組み. 言語処理学会 第 29 回年次大会 発表論文集, pp. 633–638, 2023.

A　クラスタリングの概要

BERT において用いられている 30,522 個のトークンの埋め込みを k-means 法により 1000 個のクラスタにクラスタリングした結果の概要を表 3 に示す．

表 3　単語ベクトルのクラスタリング

Type	# of clusters	# of tokens	# of tokens per cluster	Avg. norm	Avg. dist to centroid
Words	694	21,895	31.5	1.36	0.94
Subwords	248	5,561	22.4	1.61	1.06
Symbols	13	1,096	84.3	1.35	0.75
Numbers	43	973	22.6	1.45	0.60
Special	1	995	995.0	1.16	0.71
Tags	1	2	2.0	1.33	0.71
Total	1,000	30,522	30.5	1.40	0.92

表中の Type はトークンのタイプを表し，著者がマニュアルで分類した．クラスタ間の類似性を調べるために，1000 のクラスタに対して再度クラスタリングを行った．具体的には，各クラスタのセントロイド（ベクトル）に対して，（上位）クラスタ数を 7 として k-means 法を適用した．その結果，7 つの上位クラスタのうち，2 つの上位クラスタは，ほとんど Words に対応するクラスタを含んでおり，残りの 5 つの上位クラスタに含まれるクラスタは Words 以外のタイプにそれぞれ対応していた．

一方，Words タイプに属しているクラスタ 694 個のうち 65 個は，100 語以上の単語を含んでいる．すなわち，Words タイプのトークンの一部に対応するベクトルが密な部分空間に存在している．より詳細な言語的特徴に基づく分析を行う場合は，これら密に存在するベクトルは下位クラスタに再分割することが望ましい．下位クラスタへのクラスタリングにより，詳細な統語的・意味的なカテゴリを共有するクラスタへ細分類することが期待される．

B　位置ベクトルとの直交性

BERT に入力されるトークン列の状態ベクトルには，それぞれのトークンに対応する単語埋め込み $X_{emb} \in \mathbb{R}^{n \times d}$ とそのトークンの文中の位置番号に対応した位置ベクトル $X_{pos} \in \mathbb{R}^{n \times d}$ の和 $X := X_{emb} + X_{pos}$ が用いられる．位置ベクトルは文長の個数だけ存在し，BERT の base モデルの場合は 512 個が事前学習により獲得されている．

単語埋め込みと位置ベクトル（いずれも 768 次元のベクトル）について主成分分析結果（図 4）より両者は分離された部分空間に住むことがわかる．

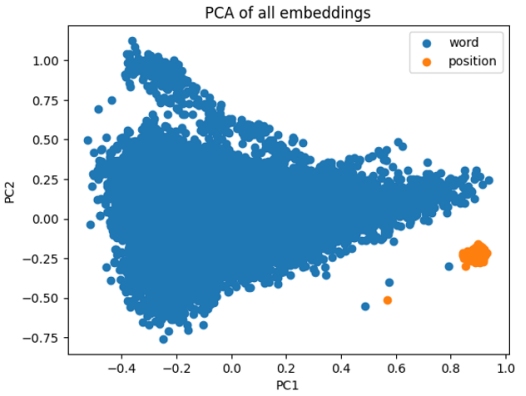

図 4　単語埋め込みと位置ベクトルの主成分分析

図 5 は，単語埋め込みと位置ベクトルの組み合わせ $30,522 \times 512$ 対のコサイン類似度の分布を示す．これは，単語埋め込みと位置ベクトルが多くの場合直交することを示す．例外的に文頭を表すタグ [CLS] に対応する単語埋め込みと，トークン位置 0 に対応する位置ベクトルのコサイン類似度は 0.99 であり，直交する空間と補空間の交線上にある．

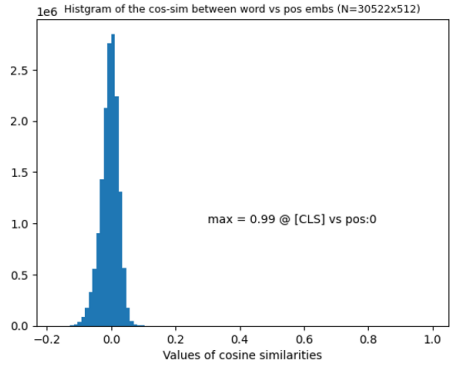

図 5　単語埋め込みと位置ベクトルの直交性

位置ベクトルは単語間の相対位置を符号化しており [18]，図 6 に示すように一部のヘッドは，skip-gram のように機能している．

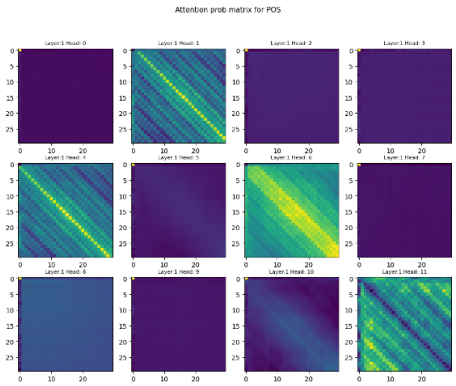

図 6　位置ベクトルから計算されるヘッド別アテンションウェイト (Layer0 の例)