Transformer の文脈を混ぜる作用と混ぜない作用

小林 悟郎 [1]　栗林 樹生 [1,2]　横井 祥 [1,3]　乾 健太郎 [1,3]
[1] 東北大学　[2]Langsmith 株式会社　[3] 理化学研究所
{goro.koba, kuribayashi, yokoi, inui}@ecei.tohoku.ac.jp

1　はじめに

近年，Transformer アーキテクチャ [1] は分野内の基盤技術となりつつあり，成功理由の解明および更なる改善に向け，Transformer を基盤としたモデル（BERT [2] など）の分析や検証が盛んに行われている [3]．Transformer は入力系列の情報を混ぜ合わせる注意機構を積み重ねた構造である．既存研究の多くは注意機構による「混ぜ合わせ」が成功の鍵であるとみなし，ある単語の表現を計算する時にどの単語の情報を用いているか，例えば依存関係や意味関係などと比較して分析している [4, 5, 6, 7, 8, 9, 10, 11, 12]．

しかし，Transformer を構成する要素は注意機構だけではない．例えば残差結合による「入力をそのまま残す作用」が存在する．もしこの影響が強ければ，既存研究が観察してきた混ぜ合わせに関する観察はモデル全体においては微々たる作用に過ぎない．また，層正規化はベクトルの正規化と拡大縮小を行い，外側から注意機構の混ぜる作用と残差結合の残す作用の強さを改変しうる．Transformer 内部において**注意機構による混ぜ合わせがどの程度支配的であるか，Transformer の主たる仕事は「混ぜる」ことなのだろうか**．本研究では残差結合と層正規化にも焦点を当ててこれらを調べる．「混ぜ合わせ」などのモデル内における情報の流れを明らかにすることは，モデルに対する理解を深めるだけでなく，モデルの予測の説明としても重要である．

実験では Transformer を基盤とした代表的なモデルとして BERT を対象に，モデル内部における混ぜ合わせの強さを調査する．実験結果から BERT の各層では，周囲のベクトルから混ぜる量よりも自身のベクトルを残す量の方が遥かに大きく，混ぜる作用ではなく残す作用が支配的であることを明らかにする．これにより，モデル内では情報がほんの少しずつ混ぜられていくことが示唆された．このような結果が得られた理由を機構ごとの作用から分析し，注意機構に比べて残差結合が強く働いていること，層正規化が特定のトークンに対応するベクトルを強く拡大していることを明らかにした．

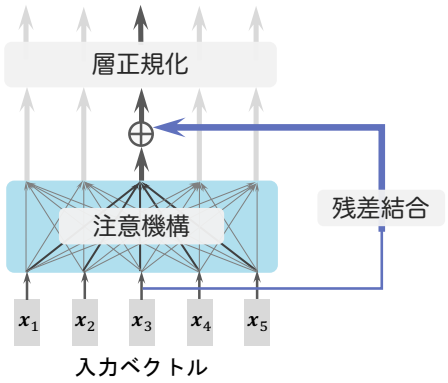

図 1　注意機構，残差結合，層正規化の概要．

本研究の貢献は以下の通りである．

- Transformer に対し，注意機構に加えて残差結合と層正規化を考慮して「混ぜ合わせの強さ」を分析する方法を提案した．
- BERT の各層では，文脈を混ぜる作用よりも自信を残す作用が支配的であることを示した．

2　準備：Transformer を構成する機構

Transformer は同じ構造を持つ層を積み重ねたアーキテクチャであり，各層は（1）注意機構，（2）残差結合，（3）層正規化，（4）順伝播型ニューラルネットワークの四つの機構から構成される．多くの先行研究はこのうち，注意機構のみを分析対象としてきた．本研究では注意機構に加えて残差結合と層正規化を考慮し，入力ベクトル $\boldsymbol{x}_j$ がこれらの機構を通過して $\boldsymbol{y}_j^{\boxed{\text{attn+res+ln}}}$ へと変化する様子を分析する．Transformer 内における注意機構，残差結合，層正規化の簡単な概要を図 1 に示す．本稿では自己注意機構を対象として提案手法の説明と実験を行うが，エンコーダーとデコーダーの間の注意機構でも同様の

議論が可能である．

2.1 注意機構（attn）

注意機構は Transformer において「混ぜる作用」を担う．以下，Transformer 内の各注意機構を慣例にならって**ヘッド**（h）と呼ぶことにする．各ヘッド h 内において，各入力ベクトル $\boldsymbol{x}_j \in \mathbb{R}^d$ は周辺の入力ベクトル全体 $\{\boldsymbol{x}_1, \ldots, \boldsymbol{x}_n\}$ を参照しながら出力ベクトル $\boldsymbol{y}_j^{(h)} \in \mathbb{R}^d$ へ更新される．具体的には，周辺のベクトル $\boldsymbol{x}_i$ を参照する度合い $\alpha_{i,j}^{(h)}$（アテンション重み）を計算し，$\boldsymbol{x}_i$ をアフィン変換したベクトル $f_j^{(h)}(\boldsymbol{x}_i)$ をこの重みで足し合わせる．

$$\boldsymbol{y}_j^{(h)} = \sum_{i=1}^{n} \alpha_{i,j}^{(h)} f_j^{(h)}(\boldsymbol{x}_i)$$

Transformer ではこのヘッドを H 個並列に並べた**マルチヘッド注意機構（attn）**が各層に組み込まれている．マルチヘッド注意機構は，以下のように各ヘッド h の出力ベクトルを足し合わせ，ベクトル $\boldsymbol{y}_j^{\boxed{\text{attn}}}$ を出力する．

$$\boldsymbol{y}_j^{\boxed{\text{attn}}} = \sum_{h=1}^{H} \boldsymbol{y}_j^{(h)}$$

2.2 残差結合（res）

残差結合（res）は，注意機構の出力 $\boldsymbol{y}_j^{\boxed{\text{attn}}}$ に，処理を加えていない入力ベクトル $\boldsymbol{x}_j$ をそのまま足し込むことでオリジナルの情報を「残す」作用を担う．注意機構と残差結合を合わせた出力 $\boldsymbol{y}_j^{\boxed{\text{attn+res}}}$ は次の通り計算される：

$$\boldsymbol{y}_j^{\boxed{\text{attn+res}}} = \boldsymbol{y}_j^{\boxed{\text{attn}}} + \boldsymbol{x}_j. \tag{1}$$

2.3 層正規化（ln）

層正規化（ln）は入ってきたベクトル $\boldsymbol{y}$ を正規化しさらに拡大縮小をほどこす：

$$\mathrm{LN}(\boldsymbol{y}) = \overline{\boldsymbol{y}} \odot \boldsymbol{\gamma} + \boldsymbol{\beta} \in \mathbb{R}^d, \quad \overline{\boldsymbol{y}} := \frac{\boldsymbol{y} - m(\boldsymbol{y})}{s(\boldsymbol{y})}.$$

まず $\boldsymbol{y} \mapsto \overline{\boldsymbol{y}}$ の計算で入力 $\boldsymbol{y}$ の要素全体が平均 0 分散 1 となるよう正規化される．$\boldsymbol{y}^{(k)}$ をベクトル $\boldsymbol{y}$ の k 番目の要素として，$m(\boldsymbol{y}) := \frac{1}{d}\sum_k \boldsymbol{y}^{(k)} \in \mathbb{R}$ は要素平均，$s(\boldsymbol{y}) := \sqrt{\frac{1}{d}\sum_k (\boldsymbol{y}^{(k)} - m(\boldsymbol{y}))^2 + \epsilon} \in \mathbb{R}$ は標準偏差，$\epsilon \in \mathbb{R}$ は数値安定のための小さな定数を表す．引き算および割り算は要素毎におこなう．

次に学習可能なアフィン変換パラメータ $\boldsymbol{\gamma}, \boldsymbol{\beta} \in \mathbb{R}^d$ によって $\overline{\boldsymbol{y}}$ の拡大縮小がおこなわれる．$\odot$ は要素積を表す．Transformer では図 1 に示すように，注意機構と残差結合が統合された各ベクトル $\boldsymbol{y}_j^{\boxed{\text{attn+res}}}$ に対して上記の計算が行われ，以下のように $\boldsymbol{y}_j^{\boxed{\text{attn+res+ln}}}$ へと更新される．

$$\boldsymbol{y}_j^{\boxed{\text{attn+res+ln}}} = \mathrm{LN}(\boldsymbol{y}_j^{\boxed{\text{attn+res}}}) = \overline{\boldsymbol{y}_j^{\boxed{\text{attn+res}}}} \odot \boldsymbol{\gamma} + \boldsymbol{\beta}$$

3 提案手法

本稿では Transformer において**注意機構による混ぜ合わせがどの程度支配的であるか**を調査するため，注意機構に加えて残差結合と層正規化を考慮した分析を行う．混ぜ具合の検証のため，それらの機構を通過して更新された埋め込みが，更新前の自身の入力とその他の入力からそれぞれどの程度ずつ集めて構成されたのかを確認する手法を提案する．具体的には，三つの機構が埋め込みを更新する計算を自身の入力に由来するベクトルとその他の入力に由来するベクトルの和に変形し，それぞれのベクトルのノルムを測ることで埋め込みの更新において他からの混ぜ合わせがどの程度行われたかを観察する．

3.1 混ぜる強さの分析

注意機構，残差結合，層正規化を通して入力ベクトル $\boldsymbol{x}_j$ が $\boldsymbol{y}_j^{\boxed{\text{attn+res+ln}}}$ へ更新される際に，自身以外の入力ベクトル $\{\boldsymbol{x}_1, \ldots, \boldsymbol{x}_{j-1}, \boldsymbol{x}_{j+1}, \ldots \boldsymbol{x}_n\}$ から集めた量（**混ぜた量**）と自身の入力ベクトル $\boldsymbol{x}_j$ から集めた量（**残した量**）を比較したい．$\boldsymbol{y}_j^{\boxed{\text{attn}}}$ および $\mathrm{LN}(\cdot)$ が以下のように分解できる[1]ことに注意すると，

$$\boldsymbol{y}_j^{\boxed{\text{attn}}} = \sum_{i=1}^{n} f_j^{\boxed{\text{attn}}}(\boldsymbol{x}_i),\ f_j^{\boxed{\text{attn}}}(\boldsymbol{x}_i) := \sum_{h=1}^{H} \alpha_{i,j}^{(h)} f_j^{(h)}(\boldsymbol{x}_i) \tag{2}$$

$$\mathrm{LN}(\sum_i \boldsymbol{z}_i) = \sum_i \frac{\boldsymbol{z}_i - m(\boldsymbol{z}_i)}{s(\sum_i \boldsymbol{z}_i)} \odot \boldsymbol{\gamma} + \boldsymbol{\beta} \tag{3}$$

$\boldsymbol{y}_j^{\boxed{\text{attn+res+ln}}}$ は次の 4 つの項に近似なしで分解できる．

$$\boldsymbol{y}_j^{\boxed{\text{attn+res+ln}}} = \sum_{i \in \{1,\ldots,n\}\setminus\{j\}} \frac{f^{\boxed{\text{attn}}}(\boldsymbol{x}_i) - m(f^{\boxed{\text{attn}}}(\boldsymbol{x}_i))}{s(\boldsymbol{y}_j^{\boxed{\text{attn}}} + \boldsymbol{x}_j)} \odot \boldsymbol{\gamma} \tag{4}$$

$$+ \frac{f^{\boxed{\text{attn}}}(\boldsymbol{x}_j) - m(f^{\boxed{\text{attn}}}(\boldsymbol{x}_j))}{s(\boldsymbol{y}_j^{\boxed{\text{attn}}} + \boldsymbol{x}_j)} \odot \boldsymbol{\gamma} \tag{5}$$

$$+ \frac{\boldsymbol{x}_j - m(\boldsymbol{x}_j)}{s(\boldsymbol{y}_j^{\boxed{\text{attn}}} + \boldsymbol{x}_j)} \odot \boldsymbol{\gamma} \tag{6}$$

$$+ \boldsymbol{\beta} \tag{7}$$

1） $\mathrm{LN}(\cdot)$ の分解に関する詳細は付録 A で説明する．

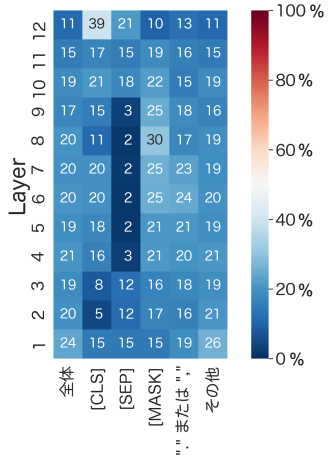

図 2　各層の注意機構・残差結合・層正規化によるベクトル更新で「混ぜ合わせ」が担う割合．

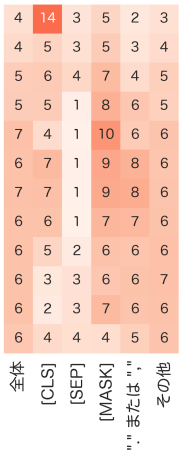

図 3　各層における注意機構が混ぜる量 $\|\sum_{i\in\{1,\dots,n\}\setminus\{j\}} f_j^{\boxed{\text{attn}}}(\boldsymbol{x}_i)\|$．

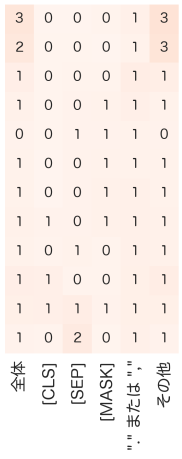

図 4　各層における注意機構が残す量 $\|f_j^{\boxed{\text{attn}}}(\boldsymbol{x}_j)\|$．

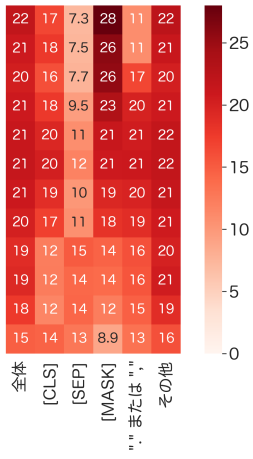

図 5　各層における残差結合が残す量 $\|\boldsymbol{x}_j\|$．

すなわち出力 $\boldsymbol{y}_j^{\boxed{\text{attn+res+ln}}}$ は

- attn 経由で混ぜる作用： $\boldsymbol{y}_{j\leftarrow\text{others}}^{\boxed{\text{attn+ln}}}$:= 項 4
- attn 経由で残す作用： $\boldsymbol{y}_{j\leftarrow j}^{\boxed{\text{attn+ln}}}$:= 項 5
- res 経由で残す作用： $\boldsymbol{y}_j^{\boxed{\text{res+ln}}}$:= 項 6
- バイアス項： $\boldsymbol{\beta}$（= 項 7）

の和に分解できる．そこで，これらの機構における「混ぜる作用」と「残す作用」をそれぞれ $\boldsymbol{y}_{j\leftarrow\text{others}}^{\boxed{\text{attn+ln}}}$ および $\boldsymbol{y}_{j\leftarrow j}^{\boxed{\text{attn+ln}}} + \boldsymbol{y}_j^{\boxed{\text{res+ln}}}$ で定義する．

注意機構をベクトルノルムで分析した Kobayashi ら [12] に従い，ベクトルの和における各ベクトルの影響の大きさをそのノルムで測る．ここでは，「混ぜた量」と「残した量」をそれぞれベクトルノルム $\|\boldsymbol{y}_{j\leftarrow\text{other}}^{\boxed{\text{attn+ln}}}\|$ および $\|\boldsymbol{y}_{j\leftarrow j}^{\boxed{\text{attn+ln}}} + \boldsymbol{y}_j^{\boxed{\text{res+ln}}}\|$ で測る．これらの大きさを比べることで，モデル内の注意機構・残差結合・層正規化部分においてどの程度の強さで混ぜ合わせが行われたかを確認する．

4　実験

Transformer を基盤とする代表的なモデルである BERT にテキストを入力し，その内部挙動について分析を行う．4.2 節では各層における「混ぜる量」と「残す量」を比べ，混ぜ合わせの強さを調べる．4.3 節および 4.4 節では，混ぜ合わせに対する注意機構，残差結合，層正規化の働きを確認した．

4.1　実験設定

モデル　事前学習済み BERT-base (uncased) を用いた．BERT-base は 12 層から構成され，その各層に図 1 で示した注意機構，残差結合，層正規化からなる構造が組み込まれている．また，追加で BERT-large についても実験を行い，結果を付録 B にまとめた．

データ　モデルへ入力するテキストデータとして，既存研究 [4] に従い Wikipedia から抽出された 992 系列を用いた．ここで BERT の事前学習の設定に合わせ，各系列において全体の 12%[2] のトークンを [MASK] トークンに置換して入力した．

4.2　混ぜる強さの分析

BERT に 992 系列を入力し，3.1 節で述べた方法でモデル内部における混ぜ合わせの強さを調べる．各層における注意機構・残差結合・層正規化部分で他の入力から混ぜた量 $\|\boldsymbol{y}_{j\leftarrow\text{others}}^{\boxed{\text{attn+ln}}}\|$ と自身の入力から残した量 $\|\boldsymbol{y}_{j\leftarrow j}^{\boxed{\text{attn+ln}}} + \boldsymbol{y}_j^{\boxed{\text{res+ln}}}\|$ を計算し，両者の合計に対する混ぜた量の割合を算出した．各層で全出力ベクトルの平均をとった結果（“全体”）に加え，対応するトークンに合わせて（1）[CLS]，（2）[SEP]，（3）[MASK]，（4）ピリオド “.” またはカンマ “,”，（5）その他の五種類に分けた結果も報告する．

結果　図 2 より，モデル内部の注意機構・残差結合・層正規化において「混ぜる作用」は “全体” で 11〜24%と小さく，「残す作用」が支配的であることがわかった．これより，BERT 内部で埋め込みが層を登って更新される際に，前の層の情報をかなり保持しながら少しずつ情報が混ぜ合わされていくことが示唆される．これは BERT 内部における中間表現は，入力トークンを高精度に判別できるほど元の情

2）BERT の事前学習では，入力系列の 20%を選び，そのうち 80%（全体の 12%）を [MASK] トークンに置換する．

報を残しているという Brunner ら [13] の報告と一致する．また，“全体” の傾向として後半層に比べて前半層の方が混ぜる作用が強いことがわかった．

トークンの種類ごとの結果に着目すると，[CLS] では混ぜる作用が最終層でのみ 39%と他よりかなり大きい傾向があった．また，[MASK] では，中盤層で 25〜30%と混ぜる作用が比較的大きかった．以上により，前半層で全体的に情報を集めて準備をし，中盤層で BERT の事前学習タスクである穴埋め予測（Masked Language Modeling）を解き，後半層でもう一つの事前学習タスクである次文予測（Next Sentence Prediction）を解いていることが示唆される．また，[CLS] および [SEP] では，それぞれ前半層および中盤層で混ぜる作用が非常に小さかった．それらの層の注意機構では，これらの特殊トークンに対してアテンション重み α を大きく割り振る一方で，あまり集めないという現象が報告されており [12]，何らかの繋がりがあると考えられる．

4.3 注意機構と残差結合の働き

前節で明らかになった「混ぜる作用ではなく残す作用が支配的」という挙動のメカニズムを調べるため，層正規化を考慮しない状態で（1）注意機構が混ぜる量，（2）注意機構が残す量，（3）残差結合が残す量を調べる．式 1 より注意機構と残差結合は，注意機構の出力ベクトル $\boldsymbol{y}_j^{\boxed{\text{attn}}}$ と，残差結合が残すベクトル $\boldsymbol{x}_j$ の足し算によって統合される．また式 2 より，注意機構の出力ベクトル $\boldsymbol{y}_j^{\boxed{\text{attn}}}$ は自身の入力 $\boldsymbol{x}_j$ から集めたベクトル $f_j^{\boxed{\text{attn}}}(\boldsymbol{x}_j)$ とその他の入力から集めたベクトル $\sum_{i\in\{1,\dots,n\}\setminus\{j\}} f_j^{\boxed{\text{attn}}}(\boldsymbol{x}_i)$ に分離できる．BERT 各層における，（1）注意機構が混ぜる量 $\|\sum_{i\in\{1,\dots,n\}\setminus\{j\}} f_j^{\boxed{\text{attn}}}(\boldsymbol{x}_i)\|$，（2）注意機構が残す量 $\|f_j^{\boxed{\text{attn}}}(\boldsymbol{x}_j)\|$，（3）残差結合が残す量 $\|\boldsymbol{x}_j\|$ を測った．

結果　図 3，4，5 より，残差結合が残す量は注意機構が混ぜる量および残す量に比べて遥かに大きく，BERT の各層において注意機構に比べて残差結合が支配的であることが分かった．また，注意機構による残す作用は小さく，残す作用のほとんどを残差結合に任せていることがわかった．

多くの先行研究で注意機構のみに着目した分析が行われ，残差結合を考慮する先行研究でも注意機構と残差結合の力関係を 1:1 と仮定していた [14] が，実際には残差結合が支配的であった．

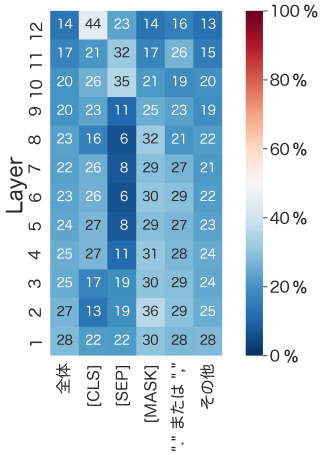

図 6　各層の注意機構・残差結合によるベクトル更新で「混ぜ合わせ」が担う割合．

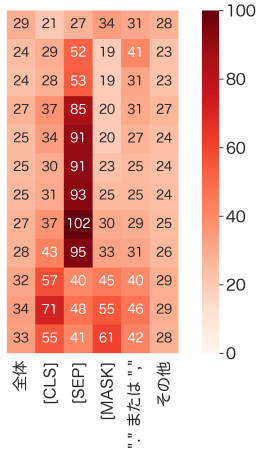

図 7　各層における層正規化を考慮した際の残差結合に由来するベクトルの大きさ $\|\boldsymbol{y}_j^{\boxed{\text{res+ln}}}\|$．

4.4 層正規化の作用

層正規化の働きについて理解を深めるため，層正規化によって「混ぜ合わせの強さ」と「残差結合が残す量」がどう変わるのかを調べた．

結果　図 2 と 6 の比較から，層正規化は全体的に混ぜ合わせの割合を下げる働きをしていることが分かった．中でも [CLS] と [SEP] に対して割合を強く下げる傾向があった．また，図 7 と 5 の比較から，層正規化は残差結合が足すベクトルを全体的に拡大していることが分かった．混ぜ合わせの割合を強く下げる [CLS] と [SEP] に対しては，特に強い拡大を行っていた．以上のように層正規化は混ぜ合わせの強さに影響を与えていた．

5 おわりに

本稿では，Transformer 内部における混ぜ合わせの強さを分析する方法を提案した．この方法を用いて BERT を分析し，各層において混ぜる作用ではなく残す作用が支配的であることを明らかにした．また，これまで盛んに分析されてきた注意機構の働きが，各層において支配的ではないことを示した．

今後は，RoBERTa[15] や fine-tuning した BERT など，より幅広くモデルを分析する．また，Transformer 内のより多くの作用を考慮に入れ，層全体やモデル全体での挙動を精緻に分析する方向性も興味深い．

謝辞．本研究は JSPS 科研費 JP19H04425, JP20J22697 の助成を受けたものです．また本研究は，JST，ACT-X，JPMJAX200S の支援を受けたものです．

参考文献

[1] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Lukasz Kaiser, and Illia Polosukhin. Attention is All you Need. In *Advances in Neural Information Processing Systems 30 (NIPS)*, pp. 5998–6008, 2017.

[2] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding. In *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) (NAACL-HLT)*, pp. 4171–4186, 2019.

[3] Anna Rogers, Olga Kovaleva, and Anna Rumshisky. A Primer in BERTology: What we know about how BERT works. *arXiv preprint arXiv:2002.12327*, 2020.

[4] Kevin Clark, Urvashi Khandelwal, Omer Levy, and Christopher D Manning. What Does BERT Look At? An Analysis of BERT's Attention. In *Proceedings of the 2019 ACL Workshop BlackboxNLP: Analyzing and Interpreting Neural Networks for NLP*, pp. 276–286, 2019.

[5] Olga Kovaleva, Alexey Romanov, Anna Rogers, and Anna Rumshisky. Revealing the Dark Secrets of BERT. *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pp. 4364–4373, 2019.

[6] Emily Reif, Ann Yuan, Martin Wattenberg, Fernanda B Viegas, Andy Coenen, Adam Pearce, and Been Kim. Visualizing and Measuring the Geometry of BERT. *Advances in Neural Information Processing Systems 32 (NIPS)*, pp. 8594–8603, 2019.

[7] Yongjie Lin, Yi Chern Tan, and Robert Frank. Open Sesame: Getting Inside BERT's Linguistic Knowledge. *Proceedings of the 2019 ACL Workshop BlackboxNLP: Analyzing and Interpreting Neural Networks for NLP*, pp. 241–253, 2019.

[8] David Mareček and Rudolf Rosa. From Balustrades to Pierre Vinken: Looking for Syntax in Transformer Self-Attentions. In *Proceedings of the 2019 ACL Workshop BlackboxNLP: Analyzing and Interpreting Neural Networks for NLP*, pp. 263–275, 2019.

[9] Phu Mon Htut, Jason Phang, Shikha Bordia, and Samuel R. Bowman. Do Attention Heads in BERT Track Syntactic Dependencies? *arXiv preprint arXiv:1911.12246*, 2019.

[10] Alessandro Raganato and Jörg Tiedemann. An Analysis of Encoder Representations in Transformer-Based Machine Translation. In *Proceedings of the 2018 EMNLP Workshop BlackboxNLP: Analyzing and Interpreting Neural Networks for NLP*, pp. 287–297, 2018.

[11] Gongbo Tang, Rico Sennrich, and Joakim Nivre. An Analysis of Attention Mechanisms: The Case of Word Sense Disambiguation in Neural Machine Translation. In *Proceedings of the Third Conference on Machine Translation (WMT): Research Papers*, pp. 26–35, 2018.

[12] Goro Kobayashi, Tatsuki Kuribayashi, Sho Yokoi, and Kentaro Inui. Attention is not only a weight: Analyzing transformers with vector norms. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pp. 7057–7075, 2020.

[13] Gino Brunner, Yang Liu, Damián Pascual, Oliver Richter, Massimiliano Ciaramita, and Roger Wattenhofer. On Identifiability in Transformers. In *8th International Conference on Learning Representations (ICLR)*, 2020.

[14] Samira Abnar and Willem Zuidema. Quantifying attention flow in transformers. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pp. 4190–4197, July 2020.

[15] Yinhan Liu, Myle Ott, Naman Goyal, Jingfei Du, Mandar Joshi, Danqi Chen, Omer Levy, Mike Lewis, Luke Zettlemoyer, and Veselin Stoyanov. RoBERTa: A Robustly Optimized BERT Pretraining Approach. *arXiv preprint arXiv:1907.11692*, 2019.

付録

A　層正規化の分配

複数のベクトル $\{z_1, \ldots, z_n\} \subseteq \mathbb{R}^d$ の和で表される $\sum_{i=1}^{n} z_i$ に対して層正規化 $LN(\cdot)$ を計算する際，式 3 のように層正規化の作用を和の中の各ベクトル z_i に分配するような形で，出力を n 個のベクトルに分解できることを文中で述べた．ここでは，この分解に至る式変形について説明する．まずは層正規化の計算を以下のように展開する：

$$\begin{aligned}\mathrm{LN}(\sum_i z_i) &= \overline{\sum_{i=1}^{n} z_i} \odot \gamma + \beta \\ &= \frac{\sum_{i=1}^{n} z_i - m(\sum_{i=1}^{n} z_i)}{s(\sum_{i=1}^{n} z_i)} \odot \gamma + \beta.\end{aligned}$$

ここで，分数部分の分子の第 2 項 $m(\sum_{i=1}^{n} z_i)$ について考える．$m(\sum_{i=1}^{n} z_i)$ は $\sum_{i=1}^{n} z_i$ の要素平均であり，以下のように変形できる：

$$\begin{aligned}m(\sum_{i=1}^{n} z_i) &= \frac{1}{d}\sum_{k=1}^{d}\left(\sum_{i=1}^{n} z_i\right)^{(k)} \\ &= \frac{1}{d}\sum_{k=1}^{d}\sum_{i=1}^{n} z_i^{(k)} \\ &= \sum_{i=1}^{n}\frac{1}{d}\sum_{k=1}^{d} z_i^{(k)} \\ &= \sum_{i=1}^{n} m(z_i).\end{aligned}$$

これに基づき，式 3 を以下のように導出できる：

$$\begin{aligned}\mathrm{LN}(\sum_i z_i) &= \frac{\sum_{i=1}^{n} z_i - \sum_{i=1}^{n} m(z_i)}{s(\sum_{i=1}^{n} z_i)} \odot \gamma + \beta \\ &= \sum_{i=1}^{n} \frac{z_i - m(z_i)}{s(\sum_{i=1}^{n} z_i)} \odot \gamma + \beta.\end{aligned}$$

B　BERT-large における実験結果

4 節で BERT-base に対して行った実験を全く同じ設定で BERT-large でも行った結果を報告する．

B.1　混ぜる強さの分析

図 8 より，BERT-large においても「混ぜる作用」は全体で 11〜28%と小さく，「残す作用」が支配的であった．また，BERT-base の結果（図 2）と比べると，最初の層と最終層を除いて BERT-large の方が「混ぜる作用」が小さく，層が増えた分だけゆっくりと混ぜ合わせが行われていると考えられる．また，[CLS] に対して強い混ぜ合わせを行うのが，BERT-base では最終層で，BERT-large では第一層という違いを確認した．

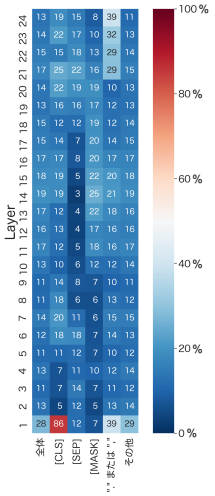

図 8　BERT-large 各層の注意機構・残差結合・層正規化によるベクトル更新で「混ぜ合わせ」が担う割合．

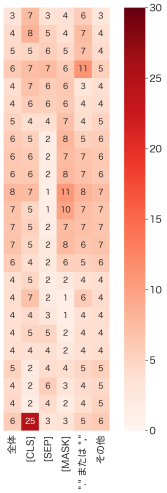

図 9　BERT-large 各層における注意機構が混ぜる量 $\|\sum_{i\in\{1,\ldots,n\}\setminus\{j\}} f_j^{\boxed{\text{attn}}}(\boldsymbol{x}_i)\|$.

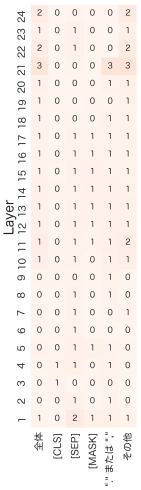

図 10　BERT-large 各層における注意機構が残す量 $\|f_j^{\boxed{\text{attn}}}(\boldsymbol{x}_j)\|$.

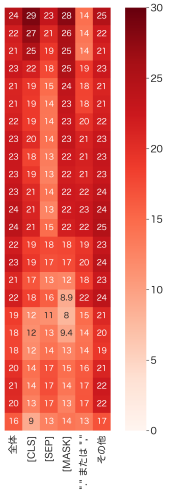

図 11　BERT-large 各層における残差結合が残す量 $\|\boldsymbol{x}_j\|$.

B.2　注意機構と残差結合の働き

図 9，10，11 より，BERT-large においても，残差結合が残す量が注意機構が混ぜる量および残す量に比べてはるかに大きく，各層において注意機構に比べて残差結合が支配的であることがわかった．

B.3　層正規化の作用

スペースの都合で結果の図は割愛するが，BERT-large においても，層正規化は全体的に混ぜ合わせの割合を下げており，特に [CLS] と [SEP] で強く割合を下げていた．ただし，第一層の [CLS] に対しては混ぜ合わせを強める作用を行っていた．また，層正規化は BERT-large においても残差結合が足すベクトルを全体的に拡大しており，特に [CLS]，[SEP]，[MASK] に対して強い拡大を行っていた．