言語モデルの学習における知識ニューロンの形成過程について

有山知希 [1]　Benjamin Heinzerling[2,1]　乾健太郎 [1,2]
[1] 東北大学　[2] 理化学研究所
tomoki.ariyama.s3@dc.tohoku.ac.jp
benjamin.heinzerling@riken.jp,　kentaro.inui@tohoku.ac.jp

概要

言語モデルが学習によって獲得した知識をどのようにモデル内に保存しているかについては様々な研究が行われているが，その中には言語モデル内に知識をエンコードしている“知識ニューロン”の存在を報告しているものがある．本研究では，そのような知識ニューロンが事前学習の経過に伴ってどのように形成されるかを調査した．調査の結果，知識ニューロンは学習中に特定の概念のみの知識を蓄えていくのではなく，学習初期の多くの概念の出力に影響を及ぼす状態から，学習が進むにつれて特定の概念の出力のみに影響を及ぼす状態に変化していく過程によって，特定の概念に特化した知識ニューロンとして形成される可能性があることが示された．

1　はじめに

事前学習済み言語モデルの中には，“[MASK] はニャーと鳴く．”という穴埋め文が与えられた時に，穴埋め部分に“猫”が入ると予測できるものがある．猫についての知識がなければこの穴埋め部分を正しく予測することはできないため，このような言語モデルには，学習によって何らかの形で猫に関する知識が保存されていると考えられる．言語モデルにおける知識については，Dai ら [1] や有山ら [2] によって，事前学習済み Transformer モデルの Feed-Forward 層（以下，“FF 層”と呼ぶ）に知識をエンコードしていると考えられるニューロン，すなわち“知識ニューロン”が存在することが報告されている．

こうした背景を踏まえ，我々はそのような知識ニューロンが事前学習中にどのように言語モデルに形成されていくかを調査することにした（図 1）．学習過程における調査を行うため，MultiBert[3] の各チェックポイントに Dai ら [1] によって提案された知識ニューロンを探す手法を適用した．その結果を学習経過に沿って分析することで，知識ニューロンの形成過程を調査した．

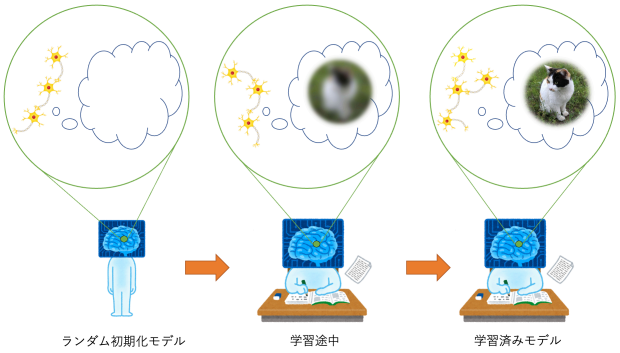

図 1　知識ニューロンは学習によってどのように形成されていくのか？

その結果，学習初期には多くの概念[1)]の出力に影響を与える状態のニューロンが，学習の経過に伴って特定の概念の出力のみに影響を与える状態に変化していくことで，特定の概念の知識に特化した知識ニューロンとして形成されていく可能性があることを確かめた．

2　手法

2.1　ニューロン

本論文におけるニューロンとは，Transformer[4] の encoder を構成する FF 層において，第一線形層の出力を活性化関数にかけたものを指す．これは，Geva ら [5] によって FF 層が key-value メモリと同様の働きをすることが報告されており，そのため FF 層には概念についての知識が保存されている可能性が高いと考えたためである．ここで，FF 層の式は入力を x，第一・第二線形層の重み・バイアス項をそれぞれ W_1, b_1, W_2, b_2 で表し，活性化関数に GELU[6] を用いると，次の式（1）の形で表される：

$$\mathrm{FF}(x) = (\mathrm{GELU}(xW_1 + b_1))W_2 + b_2 \qquad (1)$$

1)　本論文において「概念」とは，名詞や固有名詞等で表されるエンティティや，動詞や形容詞等で表される動作や性質などを表す，あらゆる単語として定義する．

式（1）中の “GELU($xW_1 + b_1$)” の部分がニューロンに対応し，その値がニューロンの活性値となる．

2.2 知識ニューロンを探すためのタスク

本節では，実験手法で使用するタスクについて説明する．知識ニューロンを探すためのタスクとして，穴埋め文の穴埋め部分を言語モデルに予測させるタスクを使用する．穴埋め文は，概念が穴埋め部分，すなわち [MASK] トークンとなるようにデータセット（3.1 節）から作成する．

また，今後の説明のために穴埋め文の呼び方を定義する．今，ある 1 つの概念 C を考えているとする．このとき，C が穴埋め部分に入る穴埋め文を，概念 C の「正例文」と呼ぶ．対して，C が穴埋め部分に入らないものは「負例文」と呼ぶ．例えば，下記は概念 “cats” の正例文と負例文の例である：

- 正例文: Kittens will grow and become [MASK].
- 負例文: Kittens will grow and [MASK] cats.

2.3 知識帰属法

本節では，Dai ら [1] によって提案された，事前学習済み言語モデルから知識ニューロンを探す手法である知識帰属法について説明する (図 2 参照)．以下では，ある 1 つの概念 C を扱うことを考え，その概念 C についての知識をエンコードしていると考えられる知識ニューロンを探す方法について述べる．

まず，言語モデル内に存在する各ニューロンのうち，“モデルが，概念 C の正例文 s に対して正しい答え C を出力する確率 $\mathrm{P}_s(\hat{w}_i^{(l)})$” に大きく影響を与えるものを探す．影響の大きさは帰属値 $\mathrm{Attr}(w_i^{(l)})$ によって測るため，その計算方法を説明する．

帰属値 $\mathrm{Attr}(w_i^{(l)})$ の計算に必要な上述の確率 $\mathrm{P}_s(\hat{w}_i^{(l)})$ は，$w_i^{(l)}$ を l 番目の FF 層の i 番目のニューロン，$\hat{w}_i^{(l)}$ をそのニューロンの活性値とすると，次の式（2）で与えられる：

$$\mathrm{P}_s(\hat{w}_i^{(l)}) = p(C|w_i^{(l)} = \hat{w}_i^{(l)}) \tag{2}$$

この確率について，Sundararajan ら [7] の “Integrated Gradients” という帰属法を用い，$\hat{w}_i^{(l)}$ を 0 から事前学習済み言語モデルにおける活性値 $\bar{w}_i^{(l)}$ まで変化させたときに，それに伴って変化する，確率 $\mathrm{P}_s(\hat{w}_i^{(l)})$ に対するニューロン $w_i^{(l)}$ の勾配 $\frac{\partial \mathrm{P}_s(\hat{w}_i^{(l)})}{\partial w_i^{(l)}}$ を積分することで，帰属値 $\mathrm{Attr}(w_i^{(l)})$ が計算される：

$$\mathrm{Attr}(w_i^{(l)}) = \bar{w}_i^{(l)} \int_{\hat{w}_i^{(l)}=0}^{\bar{w}_i^{(l)}} \frac{\partial \mathrm{P}_s(\hat{w}_i^{(l)})}{\partial w_i^{(l)}} d\hat{w}_i^{(l)} \tag{3}$$

この値が大きいほど，正例文 s に強く反応するニューロンであると判断する．この方法を用いてモデル内の全てのニューロンの s に対する帰属値を計算し，その中から帰属値の閾値 t を超えるニューロンのみを選ぶことで，正例文 s に強く反応するニューロンを選出することができる．

しかし，このように 1 つの正例文 s に反応するニューロンを選び出しても，それらは必ずしも概念 C の知識ニューロンであるとは限らない．なぜなら，式（3）の帰属値はニューロンが正例文 s に反応する度合いを表す値のため，s 内の他の単語や構文情報などに反応しているような「偽陽性の」ニューロンが選ばれているかもしれないからである[2)]．

そこで，先述した帰属値の閾値によるニューロンの選出を複数の正例文に適用する方法を採ることで，C の知識ニューロンを発見することができる：

1. 概念 C の正例文を，構文や含まれる語彙が異なるようにして複数用意する
2. 各正例文について，モデル内の全てのニューロンの帰属値を計算する
3. 各正例文について，閾値 t を超える帰属値を持つニューロンのみを選出する
4. 全ての正例文間での共有率の閾値 p を設定し，全ての正例文のうち p%以上で選出されているニューロンのみを残す

最後のステップ 4. で残ったニューロンは，各正例文で共有されている要素，すなわち “概念 C” の知識ニューロンである．

なお，ここで注意として，概念 C の知識ニューロンとは C の知識をエンコードしているニューロンのことを指すため，パラメータがランダム初期化のモデルや学習ステップ数が不十分なモデルに知識帰属法を適用して発見されるニューロンは，その学習の不十分さゆえに “知識ニューロン” とは呼べない．そのため，以下では知識帰属法によって発見されるニューロンのことを，それが知識ニューロンである場合も含め，“特定の概念の予測に寄与するニューロン” という意味で “寄与ニューロン” と呼ぶ．

2) 例えば，“Kittens will grow and become [MASK].” という穴埋め文に反応するニューロンがいくつか選出されたとしたとき，それらの中には単語 “grow” や “A and B” という構文情報に反応しているニューロンが含まれている可能性がある．

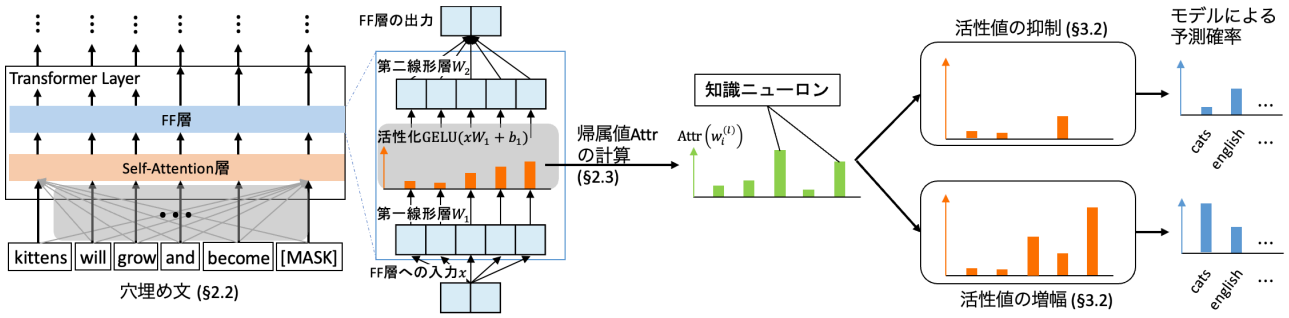

図 2　知識帰属法と活性値編集のイメージ．知識帰属法では，モデルが穴埋め文の穴埋め部分を予測する際の，FF 層における各ニューロンの活性値を用いて帰属値を計算し，それらを元に知識ニューロンを探し出す．

3　実験

3.1　設定

穴埋め文は，Generics KB[8] の中でも自然な文が提供されている GenericsKB-Best を用いて概念部分をマスクし，最終的に 4207 個の概念について作成した．このデータセットは自然かつ意味的に正しい文を大規模に提供しているだけでなく，その文のトピックである単語の情報も含んでいることから，その単語を概念として扱うことで簡単に大量の穴埋め文を作成できるため使用した．また，事前学習済み言語モデルには，MultiBert[3] の各チェックポイント（学習ステップ数 0 – 2000k）を使用した．[3)]

2.3 節の知識帰属法で用いる，各正例文における帰属値の閾値 t は，各正例文について得られた最も大きい帰属値の 0.2 倍とし，全正例文間の共有率 p は 50% に設定した．なお，実験に使用したコードはすべて公開する[4)]．

3.2　学習による知識ニューロンの形成過程

本節では，学習の経過による知識ニューロンの形成過程を調べるための実験手順について説明する．

知識ニューロンの形成過程を調査するためには，モデルの学習ステップ数の増加に伴う，寄与ニューロンの “知識ニューロンらしさ” の変化を調べれば良い．従って “知識ニューロンらしさ” についての，

(i)モデル内で知識ニューロンを抑制した場合，正例文を正解する確率は減少すると考えられる
(ii)一方で，同様の操作を行っても，負例文を正解する確率に変化はないと予想される

という前提のもと，次のような手順で実験を行う：

1. 1 つのチェックポイントに知識帰属法を適用し，各概念の寄与ニューロンを見つける
2. そのチェックポイントに正例文と負例文[5)]を予測させ，それぞれ正解する確率を測定する
3. そのチェックポイントの寄与ニューロンの活性値を 0 に抑制した上で正例文と負例文を予測させ，正解する確率を測定する (図 2 参照)
4. 手順 2. と 3. で測定した正例文についての正解確率の相対変化率[6)]，および負例文についての正解確率の相対変化率を計算することで，先述の前提をどの程度実現しているかを観察する
5. 手順 1. から 4. を，MultiBert の各チェックポイントに対して行う

なお，この一連の実験の内，手順 3. における寄与ニューロンの活性値操作を “2 倍に増幅”(図 2 参照) に変更した場合の実験も行う．この場合は，前提 (i) が “知識ニューロンを強化した場合，正例文を正解する確率は増加する” という内容に変わった上で，前提を実現している度合いを手順 4. で観察する．

4　実験結果

3.2 節の実験により得られた結果の内，寄与ニューロンの活性値を抑制した際の結果を図 3 に，増幅した際の結果を図 4 に示す．なお，プロットはその相対変化率を記録した概念の個数を表している．

まず，活性値を抑制した際の結果である図 3 を分析する．正例文を予測させた際は，学習経過に関わらず，ある正解確率の相対変化率を記録する概念の個数に大きな変化は見られない．一方で，負例文を予測させた際は，学習初期には様々な相対変化率を記録する概念が存在しているが，学習ステップ 400k

3)　より詳細な実験設定については AppendixA を参照のこと．
4)　www.github.com/tomokiariyama/knowledge-neuron-formation
5)　負例文には一律で “[MASK] is the official language of the solomon islands” を用いた．これは，実験で使用した穴埋め文の中に “english” の正例文が存在しなかったことによる．
6)　相対変化率の計算式は，Appendix B を参照のこと．

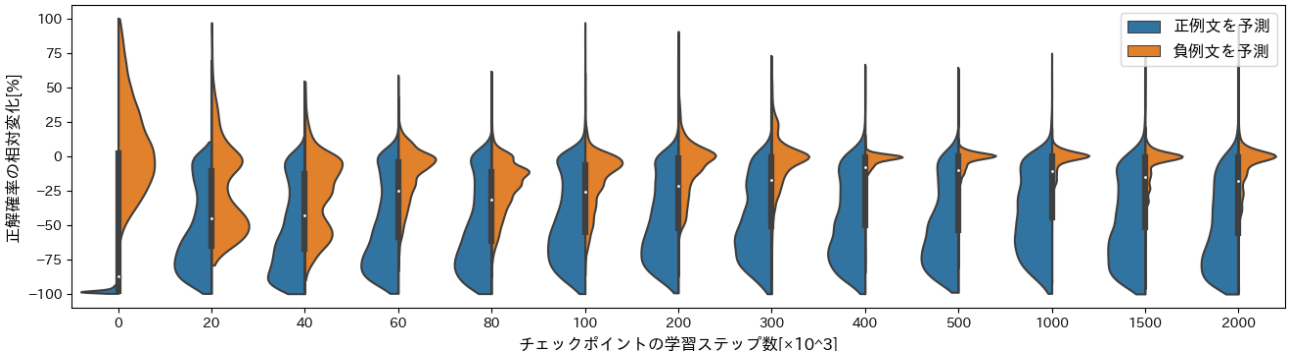

図 3　各概念の寄与ニューロンの活性値を 0 に抑制した際の，穴埋めを正解する確率の相対変化率を，モデルの学習ステップ数ごとに示したもの．プロットは，その相対変化率を記録した概念の個数を表している．

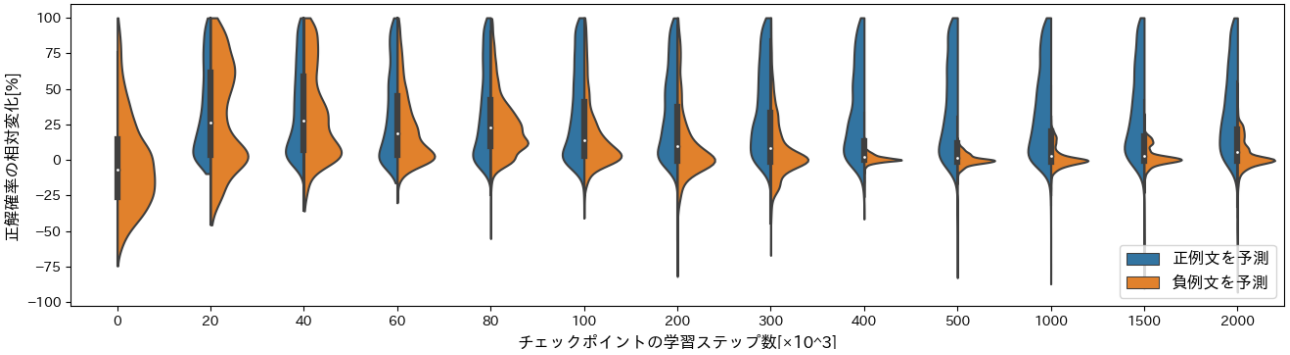

図 4　各概念の寄与ニューロンの活性値を 2 倍に増幅した際の，穴埋めを正解する確率の相対変化率を，モデルの学習ステップ数ごとに示したもの．グラフの視認性を確保するため，+100%を超える相対変化率の結果は含めていない．

を超えたあたりからほぼ全ての概念についてその変化率が 0% 付近に集中していることが分かる．

この結果は知識ニューロンの形成過程について，学習を通じて寄与ニューロンに特定の概念のみの知識が蓄積していくような過程ではなく，学習初期には寄与ニューロンが特定の概念以外も含む多くの概念の出力確率に影響を及ぼすような状態にあるが，学習が進むにつれて次第に特定の概念の出力確率以外には影響を及ぼさなくなっていくことで，特定の概念に特化した知識ニューロンを形成する，という過程を経ていることを示す可能性がある．ここで，“可能性がある” と述べたのは，この結果を説明し得る別の説が存在するためである．それは，学習初期のモデルは滑らかでないので，少数のニューロンの活性値が変化するだけでも予測に大きな影響が出ると考えられ，そのため寄与ニューロンではない，同数の他のニューロンの抑制によっても図 3 と同様の結果が得られる可能性がある，という説である．この検証については，今後の研究課題としたい．

ここまで図 3 の分析について述べたが，活性値を増幅した際の結果である図 4 でも，正例・負例文に対する正解確率の相対変化率について，上述の可能性を示唆する結果が観測された．

5　おわりに

本研究では，事前学習済み言語モデルに存在する知識ニューロンが，言語モデルの事前学習においてどのように形成されていくのか，その過程を調査した．その結果，知識ニューロンは学習初期から特定の概念のみの知識を蓄えていくのではなく，学習初期には数多くの概念の出力確率に影響を与える状態のものが，学習の進行に伴って特定の概念以外の出力確率に影響を与えなくなっていくことで，特定の概念に特化した知識ニューロンとして形成される可能性が示された．しかし，現段階では可能性に留まっており，本当にそのような形成過程を経ているかを知るためには，より詳細な実験が必要である．

さらに，ある概念についての寄与ニューロンが，学習ステップが増加しても終始同一のものである傾向が見られるかという分析も，知識ニューロンの形成過程を解明するために必須であると考えている．

また，本研究では BERT[9] モデルを研究対象としたが，その他の言語モデル，例えばより大規模なパラメータ数を持つ GPT-2[10] や T5[11] といったモデルでどのような結果が得られるかについては，興味深い研究課題として残っている．

謝辞

本研究は JST CREST JPMJCR20D2，および JSPS 科研費 21K17814 の助成を受けた．

参考文献

[1] Damai Dai, Li Dong, Yaru Hao, Zhifang Sui, and Furu Wei. Knowledge neurons in pretrained transformers. **arXiv preprint arXiv:2104.08696**, 2021.

[2] 有山知希, Benjamin Heinzerling, 乾健太郎. Transformer モデルのニューロンには局所的に概念についての知識がエンコードされている. 言語処理学会 第 28 回年次大会 発表論文集, pp. 599–603, 2022.

[3] Thibault Sellam, Steve Yadlowsky, Jason Wei, Naomi Saphra, Alexander D'Amour, Tal Linzen, Jasmijn Bastings, Iulia Turc, Jacob Eisenstein, Dipanjan Das, Ian Tenney, and Ellie Pavlick. The multiberts: Bert reproductions for robustness analysis. **arXiv preprint arXiv:2106.16163**, 2021.

[4] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Lukasz Kaiser, and Illia Polosukhin. Attention is all you need. In **Proceedings of the 31st Conference on Neural Information Processing Systems (NIPS)**, 2017.

[5] Mor Geva, Roei Schuster, Jonathan Berant, and Omer Levy. Transformer feed-forward layers are key-value memories. In **Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing (EMNLP)**, pp. 5484–5495.

[6] Dan Hendrycks and Kevin Gimpel. Gaussian error linear units (gelus). **arXiv preprint arXiv:1606.08415**, 2016.

[7] Mukund Sundararajan, Ankur Taly, and Qiqi Yan. Axiomatic attribution for deep networks. In **Proceedings of the 34th International Conference on Machine Learning**, Vol. 70, pp. 3319–3328, 2017.

[8] Sumithra Bhakthavatsalam, Chloe Anastasiades, Peter Clark. Genericskb: A knowledge base of generic statements. **arXiv preprint arXiv:2005.00660**, 2020.

[9] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. BERT: pre-training of deep bidirectional transformers for language understanding. **CoRR**, Vol. abs/1810.04805, , 2018.

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

[11] Colin Raffel, Noam Shazeer, Adam Roberts, Katherine Lee, Sharan Narang, Michael Matena, Yanqi Zhou, Wei Li, and Peter J. Liu. Exploring the limits of transfer learning with a unified text-to-text transformer. **Journal of Machine Learning Research**, Vol. 21, No. 140, pp. 1–67, 2020.

A　実験の詳細設定

実験に使用した概念は，2.3 節で述べた “全ての正例文のうち 50%以上で選出されているニューロンのみを残す” というステップを踏めるよう，GenericsKB-Best データセットから正例文が 4 つ以上作成できたものに限定した．

また，モデルとして使用した MultiBert は，公開されているシードのうち “seed_0” の各チェックポイントを使用した．

B　相対変化率の計算式

3.2 節で計算する正解確率の相対変化率は，寄与ニューロンの活性値を抑制する場合，正例・負例文共通で次の式（4）によって計算される：

$$\frac{(\text{抑制後の正解確率} - \text{抑制前の正解確率}) \times 100}{\text{抑制前の正解確率}} \tag{4}$$

寄与ニューロンの活性値を増幅する場合は，式（4）中の “抑制” を “増幅” に読み変えて計算する．