CA-003

ニューラル言語モデルが生成する自然言語文章の埋め込みベクトルの構造分析手法に関する研究

佐藤 哲 †

パーソルキャリア株式会社
テクノロジー本部 デジタルテクノロジー統括部 †

1. はじめに

自然言語処理の研究及び応用において，文章を数値で表す自然言語文章の埋め込みベクトル化は重要な課題である．近年では，ニューラルネットワークを用いた言語モデルを利用して文章の埋め込みベクトルを生成する手法が主流である．しかし，ニューラルネットワークを用いた言語モデルが出力するベクトルは次元の高さもあり，そのベクトルの成分の意味やベクトル同士の関係性など，ベクトルの構造を調べることが難しい．そこで本研究では，ニューラルネットワークを用いた言語モデルが出力する高次元の埋め込みベクトルに対し，ベクトルの幾何的な特徴を保持したまま次元を削減することで，埋め込みベクトルの構造を理解するための手法を提案する．

2. ニューラル言語モデルによる埋め込みベクトル生成

人間が使う自然言語を計算機で処理できるようにするためには，自然言語を数値で表さなければならない．自然言語の文章に対し意味を考慮して生成した数値の列を，埋め込みベクトルと呼ぶ．文章の埋め込みベクトルを生成する手法には様々なものがあるが，近年ではベクトルを出力するように訓練したニューラルネットワークによる言語モデル（以下，ニューラル言語モデル）を利用する手法が主流である．埋め込みベクトルを求めるためのニューラル言語モデルは多くの提案手法があるが [1][2]，本研究では手法のシンプルさのため，最新の手法ではないが実験・検証に現在でも広く利用されている Sentence-BERT[3] を使用する．Sentence-BERT は，入力文章に対し，標準設定では 384 次元の埋め込みベクトルを出力する．一般に，ニューラル言語モデルにより作成した埋め込みベクトルは，自然言語の多くの情報が含まれているため次元が高く，人間が理解することは難しい．そこで本研究では，高次元ベクトルに対し，人間が理解できる 2 次元のベクトル空間に高次元ベクトルを射影することで，埋め込みベクトルが持つ構造を分析する手法を提案する．高い次元のベクトルを低い次元のベクトルで表す手法は次元削減と呼ばれ多くの研究があるが，提案手法は高次元ベクトルを一つの低次元ベクトルで表すのではなく，低次元ベクトルの集合で表すところが特徴である．多数の低次元ベクトルで高次元ベクトルを表すことにより，人間が理解しやすい低次元空間で，高次元ベクトルの情報をできるだけ欠落させずに表現できることが期待される．

A Study on Structural Analysis Methods for Embedding Vectors Generated by Neural Language Models
†Tetsu R. Satoh, PERSOL CAREER CO., LTD.

3. 高次元ベクトルの低次元空間での表現

高次元ベクトルを理解するための方法の一つは，高次元ベクトルが持つ情報を適切に分析して特徴量を抽出し，低次元ベクトルで表すことである．高次元ベクトルから特徴量を抽出し，低次元ベクトルで表現する手法は次元削減と呼ばれ，多くの研究がある [5]．自然言語文章の埋め込みベクトル分析には，可視化手法としても優れている t-SNE や UMAP などの手法が使われることが多い [6]．しかし，t-SNE や UMAP などの手法は，可視化能力に優れ定性的に人間が判断するのに便利な反面，高次元ベクトルから情報を抽出できているかどうかについては検討の余地がある．t-SNE や UMAP などの手法は，高次元ベクトルからの情報抽出ではなく，高次元ベクトル群からの情報抽出のために設計されているため，ベクトル群の中のベクトルの数によって結果が変わることは避けられず，不安定な側面がある．また，一つの高次元ベクトルを一つの低次元ベクトルで表現することから，情報が欠落することは避けられない．

そこで本研究では，(1) 一つの高次元ベクトルを入力として，低次元ベクトルを出力する，(2) 一つの高次元ベクトルを多数の低次元ベクトルで表現する，というアプローチを採用する．多数の低次元ベクトルの例として，本研究では 3 次元ベクトルの集合を採用する．さらに，3 次元ベクトルの集合であれば人間が幾何的に特徴を調べることは可能であるが，大量の 3 次元ベクトルは人間による解析が困難になるため，3 次元ベクトル集合の特徴を抽出することで 2 次元ベクトル集合として高次元ベクトルを表す手法を提案する．

まず，高次元ベクトルを 3 次元空間で表現するために，Time Delay Embedding を導入する．Time Delay Embedding は，長い時系列データを有限の大きさを持つ空間に埋め込む手法の一つである [8]．Time Delay Embedding は，データ系列

$$x_k, x_{k+1}, x_{k+2}, x_{k+3}, x_{k+4}, \cdots$$

に対し，τ，L を定数とし，例えば 3 次元であれば順

に k，$k+\tau$，$k+2\tau$，$\cdots$，$k+L, k+L+\tau, k+L+2\tau$，$k+2L, k+2L+\tau, k+2L+2\tau$ とデータを取り出すことで，次のような 3 次元データ系列を生成する：

$$(x_k, x_{k+\tau}, x_{k+2\tau}), (x_{k+L}, x_{k+L+\tau}, x_{k+L+2\tau}), \cdots.$$

ここで，パラメータ τ を Time Delay，パラメータ L を Stride と呼ぶ．この手法により高次元ベクトルを 3 次元空間内に表すことができる．しかし Time Delay Embedding を埋め込みベクトルに適用すると，最大で概ね埋め込みベクトルの次元数だけの 3 次元ベクトルが生成されるため，人間が理解することが難しいという課題は解決されていない．そこで，3 次元空間のベクトルデータ集合に対し有効な手法として，位相的データ解析を適用する [7]．3 次元ベクトルデータ集合に対しパーシステンスホモロジーという量を計算することで，特徴量を 2 次元ベクトルの集合として得ることができる．パーシステンスホモロジーの計算結果を 2 次元に表した図のことをパーシステンス図と呼ぶ．パーシステンスホモロジーは，ベクトルの集合に対し，ベクトルの部分集合が作る空間の中の「穴」という幾何的な特徴を抽出する手法である．「穴」は一般化された円あるいは球であり，パーシステンスホモロジーでは穴の生成時間と消滅時間という 2 つのパラメータで扱われ，従って 2 次元データとなる．生成時間と消滅時間は，穴の位置と大きさと解釈することも可能である．ベクトルが一様乱数のように一様に分布しており特徴が無いような場合は穴の情報は抽出できないが，埋め込みベクトルのように「文章同士の意味が似ていれば埋め込みベクトル同士も似ている」というような構造が期待できる場合，ベクトルの集合に幾何的な特徴があることが予想されるため，パーシステンスホモロジーにより意味のある特徴量を抽出できる可能性がある．

以上述べたように，(1) ニューラル言語モデルにより自然言語文章の埋め込みベクトルを求める，(2) 埋め込みベクトルの Time Delay Embedding を求めることにより，埋め込みベクトルの低次元ベクトルの集合による表現を求める，(3) 低次元ベクトルの集合のパーシステンスホモロジーを計算することで，埋め込みベクトルの 2 次元ベクトル集合による表現を求める，という手続きにより，高次元の埋め込みベクトルを 2 次元空間で表す手法が提案手法である．

4. 実験例

提案手法の評価実験は，ニューラル言語モデルを用いて文章を埋め込みベクトル化し，埋め込みベクトルを低次元空間で表したとき，文章の類似度が低次元空間に反映されているかを確認することで，提案手法の妥当性を評価する．すなわち，自然言語の一対の文章をそれぞれニューラル言語モデルに入力し埋め込みベクトルを求め，提案手法により埋め込みベクトルの低次元空間での表現を求めたとき，似たような文章が入力された場合に似たような低次元空間の表現が得られるかを確認する．使用するデータは，機械学習関連のモデルの評価に使われる GLUE データセット [9] の，STS-B タスクデータとする．STS-B タスクデータは，文章の類似度を評価するためのデータセットで，一対の文章に対し意味的な類似度を表すラベルが付けられている．ラベルは 0 から 5 までの数値で表され，5 が最も文章同士が似た意味を持つことを表し，数値が小さくなるほど類似性が低いことを表す．このデータセットに対し提案手法を適用し，文章の 2 次元ベクトル集合の表現を獲得する．2 次元ベクトル集合同士の距離を計算し，類似度を表すラベルとの相関係数を計算することで，文章の意味的な類似度を保ったまま文章を低次元空間で表現できているかどうかが確認できる．類似度と距離は反比例するため，相関係数が-1 に近いほど，自然言語文章が 2 次元ベクトル集合に良好に射影できていると考えられる．

実験では，まず STS-B タスクデータに含まれる文章に対し，ニューラル言語モデルである Sentence-BERT を使って埋め込みベクトルを計算する．表 1 に，STS-B タスクデータとその埋め込みベクトルの計算例を示す．表では埋め込みベクトルの一部の成分のみが表示されているが，その中でもラベルの値が小さく，文章の意味的な類似度が低いと評価されているデータについては，ベクトルの成分の符号が違うなど，埋め込みベクトルの違いが現れていることが見て取れる．次に，Time Delay Embedding を適用し，埋め込みベクトルを 3 次元空間内のベクトル群に変換する．図 1 に．Time Dlay Embedding の適用例を示す．Sentence-BERT による 384 次元のベクトルの成分が，3 次元空間内に表されていることが分かる．上段は類似度を示すラベルが 0.5 と類似していない文章の対に対する結果で，下段が類似度を示すラベルが 5.0 で非常に類似している文章の対に対する結果である．定性的には，上段の 2 つの図に比べると，下段の 2 つの図は似通っているように思われる．この例では，Time Delay Embedding のパラメータは $\tau = 10$，$L = 10$ とした．3 次元空間内のベクトル群同士の類似度を定量的に評価するために，位相的データ解析の導入によりパーシステンスホモロジーを計算し 2 次元ベクトル集合を求めた結果を図 2 に示す．Sentence-BERT による 384 次元のベクトルが，2 次元空間内に表されていることが分かる．図 2 は，STS-B タスクデータの同一のレコードの一対の文章のうち，一つを上部，もう一つを下部とし，左が類似性が低い文章の対で，右に移るに連れ類似性が高い文章の対に対応している．従って，左側の上下の図で表される一対の文章の 2 次元ベクトル集合は似ていないことが期待され，右に移るに連れ似てくることが期待される．2 次元ベクトル集合同士が似ているかどうか定量的に判断するために

表 1: STS-B タスクデータの例

sentence1	embedding1	sentence2	embedding2	label
A man is smoking.	[0.028874, 0.035427, ...0.04330...]	A man is skating.	[-0.024052, -0.005177, ...0.016...]	0.5
Three men are playing chess.	[0.047791, -0.053223, ...-0.017...]	Two men are playing chess.	[0.039619, -0.00094, ...-0.0313...]	2.6
A man is playing a large flute.	[0.081221, -0.02136, ...0.06418...]	A man is playing a flute.	[0.043285, -0.041842, ...0.0431...]	3.8
A man is playing the cello.	[0.018254, 0.015902, ...0.04067...]	A man seated is playing the cello.	[0.020997, -0.002901, ...0.0527...]	4.25
A plane is taking off.	[0.065392, 0.02813, ...0.032704...]	An air plane is taking off.	[0.037159, 0.014117, ...0.06326...]	5.0

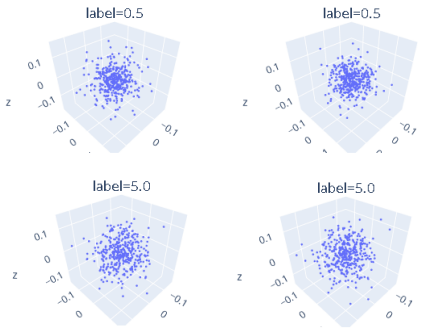

図 1: 文章データの埋め込みベクトルの 3 次元ベクトルの集合による表現例

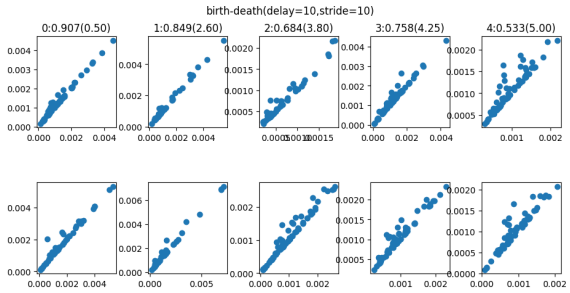

図 2: 文章データの埋め込みベクトルの 2 次元ベクトル集合による表現例

は，距離を計算する必要がある．2 次元ベクトル集合（パーシステンス図）同士の距離を計算する手法は様々なものが提案されており，ここでは Wasserstein 距離を採用する．手法の比較として，高次元ベクトルの次元削減・可視化手法として広く利用されている t-SNE[12] を使って実験する．ここでは，提案手法の実験に使用した STS-B タスクデータを用いて提案手法と同様に Sentence-BERT を用いて埋め込みベクトルを計算し，その全ての埋込ベクトルを t-SNE に入力して各ベクトルを次元削減し 2 次元での表現を求め，STS-B タスクデータの一対の文章に対応する 2 つの 2 次元ベクトル間の距離を計算し，類似度を表すラベルとの相関係数を計算する．ベクトル間の距離計算には，ユークリッド距離を用いる．

以上に述べた手続きにより，STS-B タスクデータの一対の文章に対応するラベルと，文章から求めた低次元ベクトル間の距離の，相関係数を計算した結果を表 2 に示す．Pearson の列はピアソンの相関係数を，Spearman の列はスピアマンの相関係数を表す．提案手法 (i, j) という表記は，Time Delay Embedding のパラメータが Time Delay $\tau = i$, Stride $L = j$ であることを表す．また，t-SNE(i) という表記は，t-SNE のパラメータ Perplexity の値が i であることを表している．ここでは，Time Delay Embedding については，パラメータ τ を 1, 2, 4, 8, 16 と変化させ，L についても 1, 2, 4, 8, 16 と変化させ，相関係数の絶対値が最大になる組み合わせを求めた．t-SNE についても，perplexity の値を 1, 2, 4, 8 と変化させ，相関係数の絶対値が最大となるパラメータを求めた．既に述べたように，ラベルは類似度を表し，類似度と距離は反比例するため，相関係数が-1 に近づくほど文章の意味を良好に保存した 2 次元ベクトル空間での表現が得られていると考えられる．表 2 の結果より，ピアソンの相関係数に着目した場合は提案手法 (16, 4) が最も相関係数が-1 に近く，スピアマンの相関係数に着目した場合は提案手法 (1, 8) が最も相関係数が-1 に近い．従って，いずれに着目した場合でも提案手法の方が t-SNE よりも適切にニューラル言語モデルによる埋め込みベクトルの構造を反映した低次元ベクトルを生成できていることが分かる．

以上の実験は，Python3.12 を用いて実装し，Google Cloud の a2-highgpu-1g インスタンス (メモリ 85G，GPU メモリ 40G，12vCPU) 上で実行した．パーシステンスホモロジーの計算には，

表 2: ラベルに対する相関係数

	Pearson	Spearman
提案手法 (1, 8)	-0.83	-1.0
提案手法 (16, 4)	-0.96	-0.9
t-SNE(1)	-0.87	-0.9

Homcloud[10] を用いており，Time Delay Embedding の計算には Giotto-tda[11] を用いた．

5. 考察

本研究で採用した Time Delay Embedding は，データ系列から連続的にデータを取り出す Sliding Window の拡張である．Sliding Window が近接するベクトルの成分の情報を抽出する能力があることに加え，Time Delay Embedding はよりベクトル全体の情報を抽出することができる．一般に，ニューラル言語モデルによる埋込ベクトルは，そのベクトルの成分の順番や隣接情報には余り意味が無く，ベクトル全体として自然言語文章の意味を表していると考えられている．従って，Time Delay Embedding のように順序情報に基づきベクトルの成分を抽出する必要も無く，ランダムに順序を決め，ベクトル全体から成分を抽出する方法も考えられる．実際，Time Delay Embedding におけるパラメータ Time Delay と Stride は，その値が 1 より大きくなるほどベクトルの局所的な成分情報よりも大域的な情報を重要視するようになるが，実験の結果でもパラメータを変化させた範囲の中で，大きい方が良好な結果が得られている．これらの考察から，高次元ベクトルの成分から，局所的な情報と大域的な情報の両方を考慮した特徴量抽出手法が求められていると言える．その実現方法の一つとして，乱数や重みを用いたサンプリングにより高次元ベクトルの成分から低次元ベクトル集合を構成するアプローチは有効なように思える．

6. おわりに

本研究では，ニューラル言語モデルによる自然言語文章の埋め込みベクトルが高次元で構造を分析することが難しい問題を解決するために，高次元の埋め込みベクトルを多数の低次元ベクトルで表現し，さらに位相的データ解析におけるパーシステンスホモロジーを計算することで，最終的に高次元の埋め込みベクトルを 2 次元ベクトルの集合で表現する手法を提案した．また，実験の結果より，広く利用されている t-SNE を用いて高次元ベクトルから低次元ベクトルを求めるよりも，提案手法を用いた方が良好に文章の構造を低次元ベクトルに反映できていることを示した．

今後の課題としては，提案手法を用いて高次元の埋込ベクトルの構造を分析するために，提案手法で得られた 2 次元ベクトル集合の特徴が元の埋め込みベクトルのどのような特徴に対応しているかを調べる逆解析技術の開発が挙げられる．

謝辞

本研究の一部は，パーソルキャリア株式会社テクノロジー本部デジタルテクノロジー統括部 生成系 AI・LLM 検証プロジェクトの支援による．研究の遂行における有益な助言に対し，生成系 AI・LLM 検証プロジェクトリーダーの寺本 孝太 COD グループマネージャーに深く感謝する．

参考文献

[1] A. R. Kashyap, T.-T. Nguyen, V. Schlegel, S. Winkler, S.-K. Ng and S. Poria , A Comprehensive Survey of Sentence Representations: From the BERT Epoch to the ChatGPT Era and Beyond, arXiv:2305.12641, 2023.

[2] R. Li, X. Zhao and M.-F. Moens, A Brief Overview of Universal Sentence Representation Methods: A Linguistic View, ACM Comput. Surv. Vol. 55, No. 3, Art. 56, 2023.

[3] N. Reimers and I. Gurevych, Sentence-BERT: Sentence Embeddings using Siamese BERT-Networks, Proc. EMNLP-IJCNLP, pp 3982–3992, 2019.

[4] J. Perea and J. Harer, Sliding Windows and Persistence: An Application of Topological Methods to Signal Analysis, arXiv:1307.6188, 2013.

[5] M. Ashraf, F. Anowarand, J. H. Setu, A. I. Chowdhury, E. Ahmed, A. Islam and A. Al-Mamun, A Survey on Dimensionality Reduction Techniques for Time-Series Data, Access, Vol. 11, pp. 42909-42923, 2023.

[6] Y. Wang, H. Huang, C. Rudin and Y. Shaposhnik, Understanding How Dimension Reduction Tools Work: An Empirical Approach to Deciphering t-SNE, UMAP, TriMAP, and PaCMAP for Data Visualization, arXiv:2012.04456, 2020.

[7] 平岡裕章, 位相的データ解析とパーシステントホモロジー, 数学, Vol. 68, pp. 361–380, 2016.

[8] E. Tan, S. Algar, D. Corrêa, M. Small, T. Stemler and D. Walker, Selecting embedding delays: An overview of embedding techniques and a new method using persistent homology, Chaos, Vol. 33, No. 3, pp. 032101, 2023.

[9] A. Wang, A. Singh, J. Michael, F. Hill, O. Levy and S. R Bowman, GLUE: A Multi-Task Benchmark and Analysis Platform for Natural Language Understanding, arXiv.1804.07461, 2018.

[10] I, Obayashi ,T. Nakamura and Y. Hiraoka, Persistent Homology Analysis for Materials Research and Persistent Homology Software: HomCloud, J. Physical Society of Japan, Vol. 91, No. 9, pp. 091013, 2022, doi: 10.7566/JPSJ.91.091013.

[11] G. Tauzin, U. Lupo, L. Tunstall, J. B. Pérez, M. Caorsi, A. Medina-Mardones, A. Dassatti and K. Hess, giotto-tda: A Topological Data Analysis Toolkit for Machine Learning and Data Exploration,

arXiv 2004.02551, 2020.

[12] L. Maaten and G. Hinton,Visualizing Data using t-SNE, J. Machine Learning Research, Vol. 9, No. 86, pp. 2579–2605, 2008.