プロンプトに基づくテキスト埋め込みのタスクによる冗長性の違い

塚越 駿　　笹野 遼平
名古屋大学大学院情報学研究科
tsukagoshi.hayato.r2@s.mail.nagoya-u.ac.jp
sasano@i.nagoya-u.ac.jp

概要

近年，プロンプトを与えることでタスクごとに適した埋め込み表現を出力する，プロンプトに基づくテキスト埋め込みモデルが高い性能を示している．しかし，これらのモデルはしばしば数千次元に及ぶ巨大な埋め込み表現を出力するため推論コストや保存コストに課題がある．本稿では，分類，クラスタリング，検索という 3 種のタスクに対し，事後的にこれらの埋め込みの次元数を削減した場合の性能を調査し，分類・クラスタリングタスクについては次元数を大幅に削除しても性能がほとんど損なわれないことを示す．さらに，固有次元の大きさや等方性を調査することで，大幅な次元削減が可能なタスクに適した埋め込みは冗長性が大きいことを示す．

1　はじめに

テキスト埋め込みは，類似文書検索や検索拡張生成，分類など幅広い用途で活用される自然言語処理の基盤技術である．近年になって大規模言語モデル (LLM) の高い言語理解能力を活用した埋め込みモデルの研究が盛んに行われるようになり [1, 2, 3, 4, 5, 6]，特に自然言語による指示やタスクを表現する接頭辞を付与することでタスクに適した埋め込み表現を出力できる**プロンプトに基づくテキスト埋め込みモデル**が注目されている [7, 8, 9, 10, 11, 12, 13, 14, 15]．プロンプトに基づくテキスト埋め込みモデルは多くのタスクに単一のモデルで対応できる一方，しばしば数千次元の埋め込み表現を出力するため推論コストや保存コストに課題があり，これらのモデルの出力次元数を削減することの実用上の有用性は高い．

本稿ではまず，プロンプトに基づくテキスト埋め込みモデルが特定のタスクにおいて，出力埋め込みの一部の次元のみで高い性能を達成できることを示す．特に埋め込み表現に基づくテキスト分類タスクについては，性能劣化なしに元の次元数の 1%以下まで次元削減可能であることを示す．次に，埋め込み表現の冗長性を定量的に評価することでこの要因を分析する．具体的には，タスクごとに異なるプロンプトをモデルに入力したとき，モデルから出力される埋め込み表現の固有次元や等方性がタスクごとにどのような傾向を持つかを調査する．

2　次元削減の性能への影響の比較

本節では，プロンプトに基づくテキスト埋め込みが，特定のタスクにおいて次元削減に対する高い頑健性を持ち，冗長な表現となっていることを示す．

2.1　評価タスク

多様な側面から分析を行うため，テキスト埋め込みのための包括的なベンチマークである Massive Text Embedding Benchmark (MTEB) [16] により評価する．MTEB には英語を中心に日本語やフランス語など多様な言語のデータセットが収載されているが，本稿では英語を実験と評価に用いる．MTEB の評価タスクはタスクの種別によって分類することができ，実験には分類，クラスタリング，検索の 3 つの種別からタスクを選定する[1)]．それぞれのタスク種別について以下で説明する．

分類　テキストに与えられたラベルについて，そのテキストに対応した埋め込み表現からラベルを予測する線形分類器を該当タスクの訓練セットを用いて訓練し，テストセットにおいて線形分類器を評価することで，間接的にテキスト埋め込みの良さを評価するタスクである．線形分類器の学習設定はデフォルト設定をそのまま用いる．評価指標はタスク

1)　実験に用いた各タスクの詳細は付録 A に記載する．

ごとに異なるため，それぞれのタスクごとに MTEB のデフォルトの評価指標を用いて評価する．

クラスタリング　テキストに与えられたラベルごとに，そのテキストに対応する埋め込み表現を用いてクラスタリングを行い評価するタスクである．評価指標には V-Measure [17] を用いる．

検索　検索クエリの埋め込み表現を用いて，関連文書の埋め込み表現との類似度を測り，正解文書が類似度上位に存在するかを測ることで，テキスト埋め込みの文書検索性能を評価するタスクである．評価指標には nDCG@10 を用いる．

2.2　実験モデル

プロンプトに基づくテキスト埋め込みモデルは，自然言語で記述される指示をプロンプトとして用いる**指示に基づくテキスト埋め込みモデル** [7, 8, 9, 10] と，テキスト先頭に接頭辞を付加する**接頭辞に基づくテキスト埋め込みモデル** [12, 18, 19, 13, 14, 15] の 2 種類に大別できる．一般に，指示に基づくテキスト埋め込みモデルは LLM の文脈内学習能力を利用するため，LLM を微調整して構築することが多い．接頭辞に基づくテキスト埋め込みモデルは，BERT [20] など小規模なモデルを大規模な対照学習によって微調整することで構築することが多い．本稿では，双方を実験対象に用いる．対象としたモデルの詳細は付録 B にまとめる．

指示に基づくテキスト埋め込みモデル　MTEB で高い性能を示している複数のモデルを評価に用いる．具体的には，Qwen2 7B [21] をベースに微調整を施した GTE-Qwen2，Mistral [22] をベースに GPT-4 [23] などの LLM により生成された合成データを用いて微調整を行った E5-Mistral [9]，E5-Mistral をさらに微調整した SFR-Embedding-2_R (SFR-2)，E5-Mistral と同様の方法で微調整された多言語 E5 [18] (mE5-large-inst) を用いる．これらのモデルは，タスクごとに指示を用意し，埋め込みたいテキストに指示を付加した上でモデルへ入力することで，タスクに適した埋め込み表現を出力する．

それ以外のテキスト埋め込みモデル　教師なし SimCSE [24] によって微調整された BERT [20] (Unsup-SimCSE) と，接頭辞に基づくテキスト埋め込みモデルである E5 [12] (E5-large と E5-small)，Nomic Embed [19] (nomic) を用いる．SimCSE は事前学習済み言語モデルに対して対照学習による微調整を行う手法であり，プロンプトを用いないテキスト埋め込みモデルである．E5 は対照学習によるテキスト埋め込みモデルの学習を多様なデータセットを用い大規模に行ったモデルであり，対照学習時に `query:` と `passage:` という接頭辞を埋め込みたいテキストの先頭に付加することで，検索クエリと検索対象文書の関係のように非対称性のある類似度をうまく計算できるような工夫を加えている．nomic は E5 の接頭辞を発展させ，タスクごとに 5 つの接頭辞を使い分ける．

2.3　評価手法

テキスト埋め込みモデルに対し，その埋め込み表現の次元を削減しながら評価を繰り返すことで，次元削減と性能の推移を観察する．次元削減の手法には主成分分析を用いる方法など複数の方法が考えられるが，本稿では単純に，モデルから出力されたテキスト埋め込みの先頭 d 次元を取ることで，埋め込み表現の次元を削減する[2)]．

2.4　実験結果

次元削減に伴う性能の推移について，分類タスクの性能を図 1 に，検索タスクの性能を図 2 に示す[3)]．

図 1 から，性能の推移は指示に基づくモデルとそれ以外のモデルで異なる傾向を持つことがわかる．指示に基づくテキスト埋め込みモデルの性能低下は極めて緩やかであり，特に GTE-Qwen2 や SFR-2 のテキスト分類タスクにおける性能は，**わずか 8 次元 (全体の 0.2%) でもほとんど低下しない**ことがわかる．次元数が 2 や 4 になると性能は低下したが，2 次元でもその性能は 76.34 と高く，E5-large から出力される 1024 次元の埋め込み表現をそのまま用いた場合の性能 75.69 を上回った．これに対し，E5-large などのモデルは次元削減に伴い性能が単調に減少した．一方で，図 2 から，検索タスクにおける次元削減に伴う性能低下傾向は，分類タスクと異なり，どのモデルもほとんど同じ傾向を持っていることがわかる．ただし，たとえば GTE-Qwen2 の埋め込み表

2)　埋め込み表現の先頭数次元を取得するのみで次元削減を実現できるようにする手法として，例えば Matryoshka Representation Learning [25] などの手法が存在する．本稿で次元削減をした際に得られた結果が，これらの手法に由来するものではないことを確かめるために，先頭 d 次元ではなく，ランダムに指定した d 個の次元を用いることで次元削減をした場合との比較を行った．その結果，先頭 d 次元を取得した場合と差は見られなかったため，本稿では単純に先頭 d 次元のみを用いる方法を次元削減の方法として用いた．

3)　クラスタリングタスクの性能の推移は，分類タスクと類似した傾向を示したため，紙面の都合上付録 C に記載する．

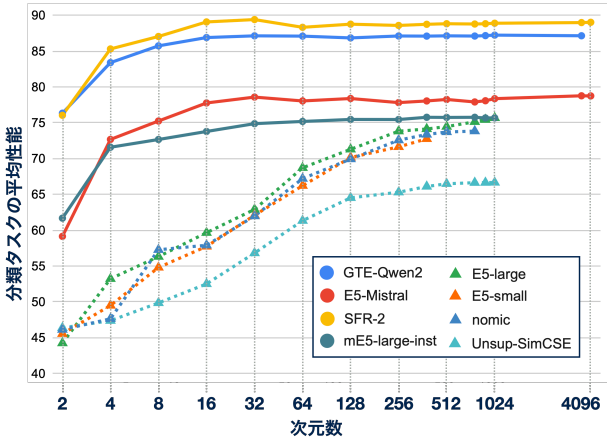

図 1　横軸に次元数を，縦軸に分類タスクの平均性能を取った図．横軸は対数軸である．丸点と実線は指示に基づくテキスト埋め込みモデルを，三角点と点線はそれ以外のモデルを表す．

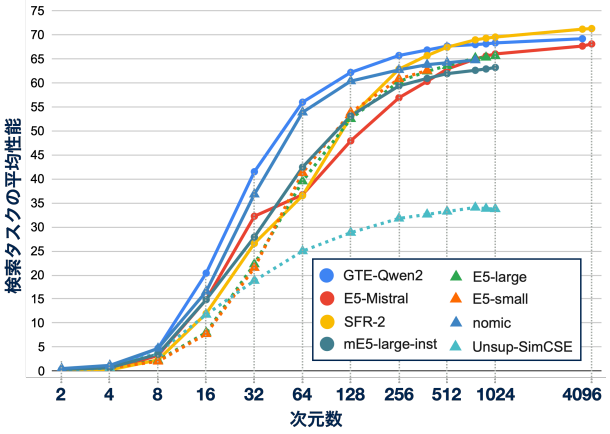

図 2　横軸に次元数を，縦軸に検索タスクの平均性能を取った図．その他は図 1 と同様である．

現を約 14%の 512 次元に削減した場合の性能低下は約 1.5 ポイントであり，依然として限定的であった．どちらのタスクにおいても，次元数を揃えたときには，大規模かつ高性能なモデルから出力された埋め込みの性能が，小規模なモデルから出力された埋め込みの性能を多くの場合で上回った．

全体として，指示に基づくテキスト埋め込みモデルは埋め込みの先頭のいくつかの次元を取るという方法でも高い性能を維持できることがわかり，埋め込み表現に冗長性がある可能性が示唆された．特に，分類タスクにおいては次元削減に対する高い頑健性が観察され，タスクごとに冗長性の異なる表現が出力されている可能性があることが分かった．

3　プロンプトによる冗長性の分析

前節で述べた埋め込み表現の振る舞いがどのような要因で生じるか考察するため，プロンプトごとの埋め込み表現の冗長性を定量的に評価する．

3.1　実験設定

プロンプトを変化させたときに，出力されたテキスト埋め込みがどの程度冗長であるかを評価する．具体的には，固有次元と IsoScore という二つの値を冗長性の指標として用い，テキスト集合に対してプロンプトを変化させたときにこれらの値がどのように変化するかを分析する．

固有次元　固有次元 (Intrinsic Dimension: ID) とは，データの本質的な構造を説明するために必要な次元数を指す値である．固有次元の推定方法は複数あるが，本稿では上田ら [26] にならい，TwoNN [27] を用いる．TwoNN は埋め込み表現の集合が与えられたときに，各点の近傍 2 点の距離の比がどれだけ急速に変化するかを測ることで，固有次元を推定する手法である．高次元空間では一般に，データが分布している空間の次元数が大きければ大きいほど，ある点の最近傍点 r_1 と準最近傍点 r_2 との距離は急速に離れていく．この「近傍点間の距離がどれほど高速に離れていくか」を，そのデータが分布する空間の次元数の推定に用いるというのが，TwoNN の直感的な解釈である．固有次元の計算には Python ライブラリの `scikit-dimension`[4] を用いる．

IsoScore　Rudman らが提案した IsoScore [28] は，埋め込み表現の等方性 (isotropy) を評価する指標である．等方性とは，埋め込み表現が特定の次元に偏らず埋め込み空間全体に均等に広がっている度合いを表す．直感的には，IsoScore は，埋め込み表現の集合が与えられたときに埋め込み表現の各次元についての分散共分散行列を計算して適切に正規化したのち，その分散共分散行列がどれほど単位行列と乖離しているかを測る指標である．

データセット　英語 Wikipedia からランダム抽出したテキスト 1 万件を入力とし，各モデル，各プロンプトごとに埋め込みを生成する．用いるモデル・指示・接頭辞は節 2 と同様である．指示に基づくテキスト埋め込みモデルは，分類など同じタスク種別でもタスクごとに異なるプロンプトを用いるため，それぞれのプロンプトを用いた場合の固有次元と IsoScore を計算した後，プロンプトの種別ごとに平均を取る．検索タスクは検索クエリと検索文書で異なる指示・接頭辞を用いることがあるため，それ

4)　https://github.com/scikit-learn-contrib/scikit-dimension

表 1　プロンプト種別・モデルごとの固有次元と IsoScore

プロンプト種別	GTE-Qwen2		E5-Mistral		SFR-2		mE5-large-inst		nomic	
	ID	IsoScore	ID	IsoScore	ID	IsoScore	ID	IsoScore	ID	IsoScore
検索クエリ	31.90	.0779	51.36	.0761	81.38	.1117	36.59	.1750	34.74	.2112
検索文書	35.94	.0813	36.69	.0332	35.07	.0555	35.58	.0752	33.78	.1930
分類	22.02	.0052	22.26	.0057	37.03	.0077	21.85	.0191	27.75	.1556
クラスタリング	10.78	.0058	13.01	.0060	16.29	.0138	17.29	.0405	26.25	.1362

表 2　モデル・プロンプトごとの固有次元と IsoScore

モデル	プロンプト	ID	IsoScore
E5-small	query:	41.57	.4419
	passage:	37.60	.3905
E5-large	query:	42.44	.2022
	passage:	38.50	.1977
Unsup-SimCSE		27.01	.1611

ぞれについて固有次元と IsoScore を計算する．よって，プロンプトの種別は，検索クエリ，検索文書，分類，クラスタリングからなる．

3.2　実験結果

表 1，表 2 に結果を示す．いずれのモデルも，固有次元の値は実際の埋め込みの次元数より大幅に小さい値だった．表 1 に注目すると，いずれのプロンプトに基づくテキスト埋め込みモデルも，分類やクラスタリングタスクのためのプロンプトを用いた場合は固有次元と IsoScore が小さく，検索クエリや検索文書のためのプロンプトを用いた場合では固有次元と IsoScore が大きくなる傾向があった．プロンプトに基づくモデル同士を比較すると，LLM に基づくモデルの方が，分類・クラスタリングタスクと検索クエリ・検索文書との固有次元・IsoScore の差が大きかった．また，指示に基づくテキスト埋め込みモデル (QTE-Qwen2, E5-Mistral, SFR-2, mE5-large-inst) は検索クエリ・文書のためのプロンプトを用いた場合と，分類・クラスタリングのためのプロンプトを用いた場合で，固有次元では平均して 10 以上，IsoScore の値は平均して 10 倍程度の大きな差があり，分類・クラスタリングタスクのための埋め込み表現は比較的冗長性の高い表現となっていた．

表 2 から，Unsup-SimCSE といったプロンプトを用いないテキスト埋め込みモデルや E5 は，固有次元・IsoScore ともに比較的大きな値を取ることがわかる．特に，E5 は接頭辞によらず大きな値を取り，その固有次元は LLM に基づくテキスト埋め込みよりも大きな値となっていることが多い．E5 は検索クエリやテキスト分類など多様なタスクに `query:` の接頭辞を用いるため，様々な情報を埋め込み・保持するために冗長性の小さな表現を出力している可能性が示唆される．

検索タスクは，文・文書同士の繊細な意味関係を捉える必要があるため，保持しておくべき情報が多いタスクであるが，分類・クラスタリングタスクは，それぞれのテキストの意味を捨象し，特定のクラスの情報のみを持つような埋め込み表現を出力する必要があるタスクである．全体として，プロンプトに基づく埋め込みモデルはこれらのタスクの性質を考慮し，分類・クラスタリングタスクでは固有次元が小さく冗長な表現を出力しているのに対し，検索タスクでは固有次元が大きく冗長性の少ない表現を出力していることが分かった．

4　まとめ

本稿では，プロンプトに基づくテキスト埋め込みモデルが出力する高次元な埋め込み表現は，埋め込み表現の一部の次元のみを利用するという簡単な次元削減でも，高い性能を維持できることを示した．特に分類タスクでは，数次元程度への大胆な次元削減を施しても十分な性能を維持することができることを示し，この振る舞いが指示に基づくモデル特有のものであることを確認した．固有次元と IsoScore を用いた分析の結果，プロンプトに基づくテキスト埋め込みモデルは，プロンプトごとに異なる性質の埋め込み表現を出力していることがわかった．具体的には，検索タスクのように詳細な類似度が重視されるタスクでは埋め込み表現の固有次元は大きくなり，埋め込み空間上で等方的に分布する一方，分類やクラスタリングタスクのための埋め込み表現の固有次元は小さく，埋め込み空間上の偏った場所に分布する異方的な埋め込みを出力する傾向があることを確認した．固有次元・IsoScore の大きさと次元削減に伴う性能推移の傾向には関連が見られ，分類・クラスタリングタスクでは冗長性の高い埋め込み表現が，検索タスクでは冗長性の低い埋め込み表現が出力される傾向にあることが分かった．

謝辞

本研究は JSPS 科研費 23KJ1134，24H00727 の助成を受けたものです.

参考文献

[1] Niklas Muennighoff. SGPT: GPT Sentence Embeddings for Semantic Search. **arXiv:2202.08904**, 2022.

[2] Jianmo Ni, Chen Qu, Jing Lu, Zhuyun Dai, Gustavo Hernandez Abrego, Ji Ma, Vincent Zhao, Yi Luan, Keith Hall, Ming-Wei Chang, and Yinfei Yang. Large Dual Encoders Are Generalizable Retrievers. In **Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing (EMNLP)**, pp. 9844–9855, 2022.

[3] Jianmo Ni, Gustavo Hernandez Abrego, Noah Constant, Ji Ma, Keith Hall, Daniel Cer, and Yinfei Yang. Sentence-T5: Scalable Sentence Encoders from Pre-trained Text-to-Text Models. In **Findings of the Association for Computational Linguistics: ACL 2022**, pp. 1864–1874, 2022.

[4] Chihiro Yano, Akihiko Fukuchi, Shoko Fukasawa, Hideyuki Tachibana, and Yotaro Watanabe. Multilingual Sentence-T5: Scalable Sentence Encoders for Multilingual Applications. In **Proceedings of the 2024 Joint International Conference on Computational Linguistics, Language Resources and Evaluation (LREC-COLING)**, pp. 11849–11858, 2024.

[5] Jacob Mitchell Springer, Suhas Kotha, Daniel Fried, Graham Neubig, and Aditi Raghunathan. Repetition Improves Language Model Embeddings, 2024.

[6] Ting Jiang, Shaohan Huang, Zhongzhi Luan, Deqing Wang, and Fuzhen Zhuang. Scaling Sentence Embeddings with Large Language Models. In **Findings of the Association for Computational Linguistics: EMNLP 2024**, 2024.

[7] Hongjin Su, Weijia Shi, Jungo Kasai, Yizhong Wang, Yushi Hu, Mari Ostendorf, Wen-tau Yih, Noah A. Smith, Luke Zettlemoyer, and Tao Yu. One Embedder, Any Task: Instruction-Finetuned Text Embeddings. In **Findings of the Association for Computational Linguistics: ACL 2023**, pp. 1102–1121, 2023.

[8] Akari Asai, Timo Schick, Patrick Lewis, Xilun Chen, Gautier Izacard, Sebastian Riedel, Hannaneh Hajishirzi, and Wen-tau Yih. Task-aware Retrieval with Instructions. In **Findings of the Association for Computational Linguistics: ACL 2023**, pp. 3650–3675, 2023.

[9] Liang Wang, Nan Yang, Xiaolong Huang, Linjun Yang, Rangan Majumder, and Furu Wei. Improving Text Embeddings with Large Language Models. In **Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (ACL)**, pp. 11897–11916, 2024.

[10] Chankyu Lee, Rajarshi Roy, Mengyao Xu, Jonathan Raiman, Mohammad Shoeybi, Bryan Catanzaro, and Wei Ping. NV-Embed: Improved Techniques for Training LLMs as Generalist Embedding Models. **arXiv:2405.17428**, 2024.

[11] Jinhyuk Lee, Zhuyun Dai, Xiaoqi Ren, Blair Chen, Daniel Cer, Jeremy R. Cole, Kai Hui, Michael Boratko, Rajvi Kapadia, Wen Ding, Yi Luan, Sai Meher Karthik Duddu, Gustavo Hernandez Abrego, Weiqiang Shi, Nithi Gupta, Aditya Kusupati, Prateek Jain, Siddhartha Reddy Jonnalagadda, Ming-Wei Chang, and Iftekhar Naim. Gecko: Versatile Text Embeddings Distilled from Large Language Models. **arXiv:2403.20327**, 2024.

[12] Liang Wang, Nan Yang, Xiaolong Huang, Binxing Jiao, Linjun Yang, Daxin Jiang, Rangan Majumder, and Furu Wei. Text Embeddings by Weakly-Supervised Contrastive Pre-training. **arXiv:2212.03533**, 2022.

[13] Zehan Li, Xin Zhang, Yanzhao Zhang, Dingkun Long, Pengjun Xie, and Meishan Zhang. Towards General Text Embeddings with Multi-stage Contrastive Learning. **arXiv:2308.03281**, 2023.

[14] Shitao Xiao, Zheng Liu, Peitian Zhang, and Niklas Muennighoff. C-Pack: Packaged Resources To Advance General Chinese Embedding. In **The 47th International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR)**, 2024.

[15] Hayato Tsukagoshi and Ryohei Sasano. Ruri: Japanese General Text Embeddings. **arXiv:2409.07737**, 2024.

[16] Niklas Muennighoff, Nouamane Tazi, Loic Magne, and Nils Reimers. MTEB: Massive Text Embedding Benchmark. In **Proceedings of the 17th Conference of the European Chapter of the Association for Computational Linguistics (EACL)**, pp. 2014–2037, 2023.

[17] Andrew Rosenberg and Julia Hirschberg. V-Measure: A Conditional Entropy-Based External Cluster Evaluation Measure. In **Proceedings of the 2007 Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning (EMNLP-CoNLL)**, pp. 410–420, 2007.

[18] Liang Wang, Nan Yang, Xiaolong Huang, Linjun Yang, Rangan Majumder, and Furu Wei. Multilingual e5 text embeddings: A technical report. **arXiv:2402.05672**, 2024.

[19] Zach Nussbaum, John X. Morris, Brandon Duderstadt, and Andriy Mulyar. Nomic Embed: Training a Reproducible Long Context Text Embedder. **arXiv:2402.01613**, 2024.

[20] 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 (NAACL)**, pp. 4171–4186, 2019.

[21] Qwen2 Team. Qwen2 Technical Report. **arXiv:2407.10671**, 2024.

[22] Mistral Team. Mistral 7B. **arXiv:2310.06825**, 2023.

[23] GPT-4 Team. GPT-4 Technical Report. **arXiv:2303.08774**, 2024.

[24] Tianyu Gao, Xingcheng Yao, and Danqi Chen. SimCSE: Simple Contrastive Learning of Sentence Embeddings. In **Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing (EMNLP)**, pp. 6894–6910, 2021.

[25] Aditya Kusupati, Gantavya Bhatt, Aniket Rege, Matthew Wallingford, Aditya Sinha, Vivek Ramanujan, William Howard-Snyder, Kaifeng Chen, Sham M. Kakade, Prateek Jain, and Ali Farhadi. Matryoshka Representation Learning. In **Neural Information Processing Systems (NeurIPS)**, 2022.

[26] 上田亮, 横井祥. 言語の固有次元を測る. 言語処理学会 第 30 回年次大会, 2024.

[27] Elena Facco, Mariad' Errico, Alex Rodriguez, Alessandro Laio. Estimating the intrinsic dimension of datasets by a minimal neighborhood information. **Scientific Reports**, Vol. 7, , 2017.

[28] William Rudman, Nate Gillman, Taylor Rayne, and Carsten Eickhoff. IsoScore: Measuring the Uniformity of Embedding Space Utilization. In Smaranda Muresan, Preslav Nakov, and Aline Villavicencio, editors, **Findings of the Association for Computational Linguistics: ACL 2022**, pp. 3325–3339, 2022.

表 3　評価に用いたタスクと対応する指示

タスク名	指示
AmazonCounterfactualClassification	Classify a given Amazon customer review text as either counterfactual or not-counterfactual
AmazonPolarityClassification	Classify Amazon reviews into positive or negative sentiment
AmazonReviewsClassification	Classify the given Amazon review into its appropriate rating category
ImdbClassification	Classify the sentiment expressed in the given movie review text from the IMDB dataset
ToxicConversationsClassification	Classify the given comments as either toxic or not toxic
ArxivClusteringS2S	Identify the main and secondary category of Arxiv papers based on the titles
RedditClustering	Identify the topic or theme of Reddit posts based on the titles
StackExchangeClustering	Identify the topic or theme of StackExchange posts based on the titles
MIRACLRetrievalHardNegatives	Given a question, retrieve Wikipedia passages that answer the question
QuoraRetrievalHardNegatives	Given a question, retrieve questions that are semantically equivalent to the given question
HotpotQAHardNegatives	Given a multi-hop question, retrieve documents that can help answer the question
DBPediaHardNegatives	Given a query, retrieve relevant entity descriptions from DBPedia
NQHardNegatives	Given a question, retrieve Wikipedia passages that answer the question
MSMARCOHardNegatives	Given a web search query, retrieve relevant passages that answer the query

表 4　各モデルの設定やプロンプトの形式

モデル名	HuggingFace	プロンプト	次元数	パラメータ
GTE-Qwen2	Alibaba-NLP/gte-Qwen2-7B-instruct	指示	3584	7.61B
E5-Mistral	intfloat/e5-mistral-7b-instruct	指示	4096	7.11B
SFR-2	Salesforce/SFR-Embedding-2_R	指示	4096	7.11B
mE5-large-inst	intfloat/multilingual-e5-large-instruct	指示	1024	560M
nomic	nomic-ai/nomic-embed-text-v1.5	接頭辞 (5 種)	768	137M
E5-small	intfloat/e5-small-v2	接頭辞 (2 種)	384	33M
E5-large	intfloat/e5-large-v2	接頭辞 (2 種)	1024	335M
Unsup-SimCSE	princeton-nlp/unsup-simcse-bert-large-uncased	なし	1024	335M

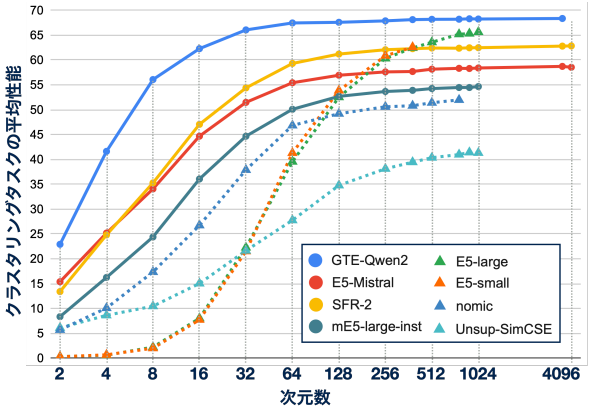

図 3　横軸に次元数を，縦軸にクラスタリングタスクの平均性能を取った図．その他は図 1 と同様である．

A　各タスクの詳細

実験に用いた MTEB 中のタスクと，指示に基づくテキスト埋め込みモデルを用いる場合にそれぞれのタスクで利用した指示を表 3 に示す．

B　評価対象モデル

本稿で評価実験に用いたモデルを表 4 に一覧する．なお，Wang ら [9] にならい，指示に基づくテキスト埋め込みモデルを検索タスクに適用する際には，検索クエリに対しては指示をプロンプトに含めるが，検索文書に対しては空文字列をプロンプトとする．接頭辞について，E5-large と E5-small では，検索文書には `passage:` を，それ以外のテキストに対しては `query:` を用いる．nomic では，検索クエリには `search_query:` を，検索文書には `search_document:` を，分類には `classification:` を，クラスタリングには `clustering:` を用いる．

C　クラスタリングタスクの性能

図 3 に，次元削減に伴うクラスタリングタスクの性能の推移を示す．大幅な次元削減を行う場合の性能低下幅は分類タスクより大きいが，LLM に基づくテキスト埋め込みモデルは，次元数を 128 次元程度に削減してもほとんど性能が低下していないことがわかる．一方で，E5-large は次元数を 12.5%の 128 次元へ削減すると，約 13 ポイント性能が低下した．E5-large，E5-small については次元削減に伴う性能低下が大きく，32 次元を境に Unsup-SimCSE の性能を下回った．表 1，表 2 より，E5-large と E5-small のクラスタリングタスク用の埋め込み表現の固有次元，IsoScore は他のどのモデルよりも大きな値を取っており，クラスタリングタスクにおける性能の急速な低下との関連が示唆される．