訓練不要な条件付きテキスト埋め込み

山田 康輔　　張 培楠
株式会社サイバーエージェント
{kosuke_yamada, zhang_peinan}@cyberagent.co.jp

概要

条件付きテキスト埋め込みは、特定の側面に焦点を当てたテキストの埋め込み表現であり、与えられた条件に基づくテキスト同士の類似度の算出を可能にする。従来手法は、大規模な訓練データによる指示学習や意味的テキスト類似度算出タスクによる微調整が求められ、開発コストが高い。そこで本研究では、生成型 LLM をテキストエンコーダとして条件付き一語制約プロンプトを用いる、訓練不要で高品質な条件付きテキスト埋め込み PonTE を提案する。条件付き意味的類似度テキスト類似度とテキストクラスタリングによる二つの実験を通じて、提案手法は追加の訓練なしで従来手法以上の性能を達成することを示す。

T_1:	This camera is one of my favorites.
T_2:	This smartphone cannot capture high-quality images.
T_3:	Best fish I have ever had.

表 1　レビューテキストの例

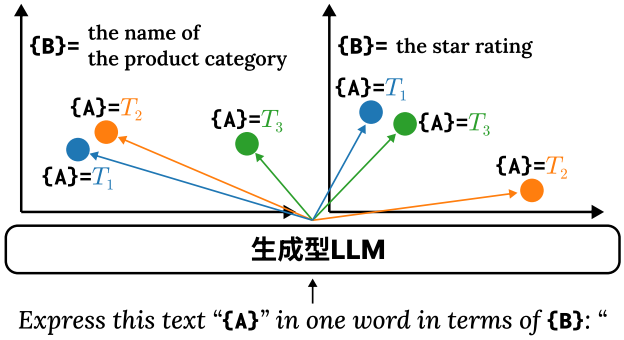

図 1　PonTE による条件付きテキスト埋め込みの可視化例。T_1、T_2、T_3 は表 1 の事例に対応する。

1　はじめに

テキスト間の類似度は類似文検索や文書クラスタリングなどにおいて重要な役割を果たす NLP タスクの一つであり [1, 2, 3]、効率的かつ類似度の算出を実現するために、テキストの埋め込み表現が一般的に使用される。ただし、従来のテキスト埋め込み手法 [4, 5, 6, 7] は、一つのテキストに対して一つの汎用的な埋め込み表現を生成するものが主要であるが、テキストには多様な側面があることから、想定する類似度の算出が困難な場合がある。たとえば、表 1 にあるようなレビューテキストの場合、T_1 と T_2 は類似したカテゴリの商品について言及しているものの、その感情極性は異なる。その一方で、T_1 と T_3 は異なるカテゴリの商品について言及しているが、どちらも肯定的な評価をしている。これらの事例では、着目する側面という条件を与えることなく、類似度の高低を判断することは難しい。

このような背景から、特定の側面に焦点を当ててテキストを埋め込む「条件付きテキスト埋め込み手法」が提案されている [8, 9]。しかし、これらの従来手法は、埋め込みモデルを訓練するための特定の条件に関するテキスト同士の類似度をアノテーションしたデータセットを必要とし、現在、英語の画像キャプションデータに付与されたデータしか存在しないため、分野や言語を超えた NLP タスクへの適用は容易ではない。また、テキスト埋め込み用に指示学習された手法も、タスクごとに指示文を与えるため条件付きテキスト埋め込みとして活用できるが、大規模な訓練データを整備し、長時間に渡って訓練する必要があり、開発コストが高い [10, 11, 12]。

そこで本研究では、訓練不要で様々な分野や言語に適用可能な条件付きテキスト埋め込み手法 **PonTE** (**P**rompt-based **Con**ditional **T**ext **E**mbedding) を提案する。概要を図 1 に示す。Jiang ら [13] が提案した一語制約プロンプトを拡張した条件付き一語制約プロンプトを用い、与えられた条件を満たすようなテキスト埋め込みを、Mistral [14] や Llama-3 [15] などの生成型 LLM から生成する。図 1 は、表 1 のテキストに対して PonTE を適用したときの埋め込み空間を可視化した結果を示し、条件に応じて事例間の距離が変わっていることを示している。

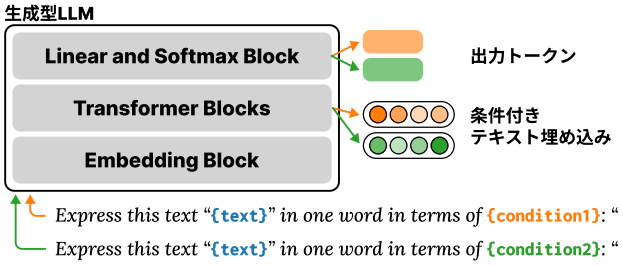

図 2　PonTE の概要

2　PonTE

従来の条件付きテキスト埋め込み手法は、特定の条件に関してテキスト間の類似度がアノテーションされたデータを用いて微調整する必要があり、分野や言語に依存しない汎用的な手法は探索されていない。また、テキスト埋め込み用に指示学習された手法も開発コストが高いため実現は容易ではない。そこで、開発コストの低い汎用的な手法を実現するため、プロンプトベースの条件付きテキスト埋め込み手法 PonTE を提案する。

PonTE の手法の概要を図 2 に示す。PonTE では、PromptEOL と同様にプロンプトを生成型 LLM に入力し、プロンプトの最終トークンに位置する Transformer ブロックの中間表現をテキスト埋め込みに用いる。PromptEOL では一語制約プロンプトを用いて情報を圧縮するが、PonTE では一語制約プロンプトを拡張した「条件付き一語制約プロンプト」を使用する。条件付き一語制約プロンプトには、情報を圧縮するための一語制約に加えて、圧縮する方向性を指定するための条件制約を導入する。たとえば、`{text}`を埋め込むテキスト本文、`{condition}`を条件として、条件付き一語制約プロンプトは「*Express this text* "`{text}`" *in one word in terms of* `{condition}`*: "*」になる。

また、プロンプトを生成型 LLM に入力し、Linear and Softmax ブロックの出力を通じて生成される一語を PonTE の埋め込みの解釈に利用する。この一語は「"」が生成されるまでの出力トークンを連結されたものを指し、予測結果の理解やプロンプトの決定に有用である。

3　実験: C-STS

PonTE が、条件に沿ったテキスト埋め込み表現が生成できているかを確認するために条件付き意味的テキスト類似度 (Conditional Semantic Textual Similarity; C-STS) の実験を行う。

手法	r_s	r_p
sup-SimCSE$_{\text{large}}$	3.4	4.1
GTE$_{\text{Qwen2-7B-Inst}}$	33.5	33.9
E5$_{\text{Mistral-7B-Inst}}$	**34.8**	**34.6**
unsup-SimCSE$_{\text{large}}$	2.3	1.7
PonTE$_{\text{Mistral-7B}}$	21.6	21.0
PonTE$_{\text{Mistral-7B-Inst}}$	30.6	28.9
PonTE$_{\text{Llama-3-8B}}$	21.7	19.7
PonTE$_{\text{Llama-3-8B-Inst}}$	**37.1**	**33.6**
PonTE$_{\text{Llama-3-70B}}$	11.3	10.9
PonTE$_{\text{Llama-3-70B-Inst}}$	35.1	31.0

表 2　C-STS の実験結果。上段が教師あり学習の手法、下段が教師なし学習の手法である。

3.1　実験設定

データセットは、Deshpande ら [8] によって作成された C-STS データセットを用いた[1)]。各レコードは、二つのテキスト、条件、類似度ラベルから構成されている。予測時に使用した類似度は二つのテキストにおける埋め込み表現のコサイン類似度で、類似度ラベルと予測類似度から算出されるスピアマン順位相関係数 (r_s) とピアソン積率相関係数 (r_p) を指標として評価した。

PonTE でテキストエンコーダに用いる LLM として、Mistral 7B のベースモデルと指示学習モデル、Llama-3 8B のベースモデルと指示学習モデルを用いた[2)]。プロンプトテンプレートは複数の候補の中から各モデルで最もスピアマン順位相関係数の高いものを使用した[3)]。PonTE との比較手法として、追加の訓練を必要とする、指示学習された Qwen2 7B [16] と Mistral 7B を元に、それぞれテキスト埋め込み用にも指示学習された GTE [11] と E5 [17, 12] を用いた。また、SimCSE [7] の教師あり学習をしたモデルと教師なし学習をしたモデルを用いて、埋め込み対象のテキストと条件のテキストを連結する形式でモデルに入力して埋め込み表現を取得した。

3.2　実験結果

実験結果を表 2 に示す。PonTE は、エンコーダに用いた LLM に関わらず高い性能を示した。特に、PonTE$_{\text{Llama-3-8B-Inst}}$ が高いスコアを達成しており、スピアマン順位相関係数では追加訓練を必要とする手

1)　データセットの統計は Appendix C に示す
2)　使用したモデルは Appendix D に示す
3)　プロンプトテンプレートの候補は Appendix A に示す

	テキスト 1	テキスト 2	条件	ラベル	予測	生成語 1	生成語 2
(a)	T_{a1}: A group of elephants of different sizes walking together on dirt with a rock formation and trees in the background.	T_{a2}: One elephant is squirting water out of its mouth and the other is putting water into its mouth.	C_{a1}: the physical actions	1.0	1.30	Walking	water-squirting
			C_{a2}: the animal	5.0	4.77	Elephants	Elephant
(b)	T_{b1}: A man in a shirt and tie with his hands in his pockets leaning against a wall.	T_{b2}: The man is wearing a dress coat, suit and tie, but not dress pants.	C_{b1}: the attire of the person	2.0	4.52	Formal	Formal
			C_{b2}: the gender of the person	5.0	4.64	Male	Male

表 3 $\text{PonTE}_{\text{Llama-3-8B-Inst}}$ の出力例。「予測」は予測類似度を 0.5 から 5.5 で Min-Max 正規化を適用した値を示し、「生成語 1」と「生成語 2」は「テキスト 1」と「テキスト 2」のときの生成された一語を示す。

法を上回るなど、多大な開発コストをかけることなく性能の高い条件付きテキスト埋め込み手法を実現できることが示唆される結果となった。

共に Mistral 7B を元にした $\text{E5}_{\text{Mistral-7B-Inst}}$ と $\text{PonTE}_{\text{Mistral-7B-Inst}}$ を比較すると、E5 が PonTE を上回っており、テキスト埋め込み用の指示学習が効果的であることがわかる。ただし、$\text{E5}_{\text{Mistral-7B-Inst}}$ は、15 万のユニークな指示文を用いて LLM で生成された 50 万の事例と、人手で作成された質問応答や検索のデータセットから収集した 180 万の事例を用いて訓練しており、その開発は容易ではない。

PonTE において、LLM のベースモデルと指示学習されたモデルによる手法を比較すると、後者が一貫して高いスコアを示した。これは、指示学習によってプロンプトの指示に従う能力が向上したことによるものだと考えられる。また、$\text{PonTE}_{\text{Llama-3-8B-Inst}}$ と $\text{PonTE}_{\text{Llama-3-70B-Inst}}$ では、$\text{PonTE}_{\text{Llama-3-70B-Inst}}$ の方が一般的に性能が高いとされているにも関わらず、$\text{PonTE}_{\text{Llama-3-8B-Inst}}$ の方がスコアが高かった。Jiang ら [13] の様々なパラメータ数のエンコーダを元にした PromptEOL を用いて STS の実験を行った結果においても、スケールの増大は性能に直接寄与しない傾向があり、実験結果はこの傾向と一致する。

3.3 分析

PonTE による条件付きテキスト埋め込みの挙動を明らかにするため、プロンプトを入力して生成された一語と埋め込みの二次元への射影結果を分析した。表 3 に $\text{PonTE}_{\text{Llama-3-8B-Inst}}$ の出力例を示す。表には、二つのテキストに対して、二つの条件における類似度ラベルと予測類似度、生成された一語を示す。表 3 (a) より、PonTE は条件ごとに類似度ラベルと近い予測類似度を示していることが確認できる。生成された一語に関してもテキスト中の条件に関連

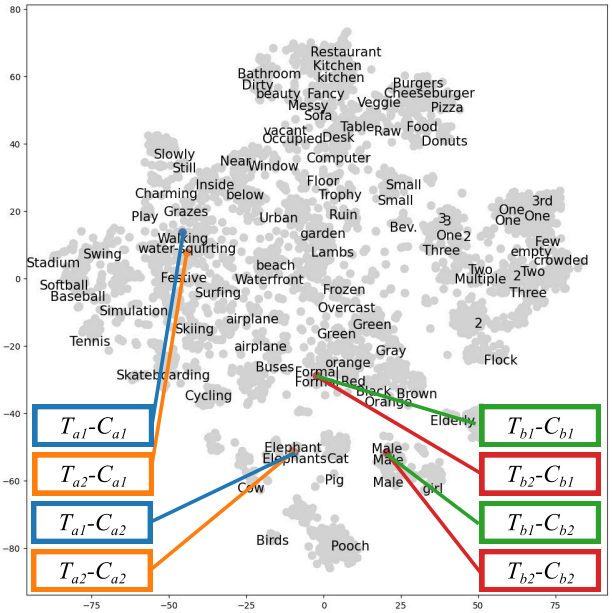

図 3 C-STS データセットにおける PonTE による埋め込みの二次元マッピング

した語を生成しており、さらに、生成された語の意味が似ているものは類似度が高く、そうではないものは低くなっている。この結果から、生成された語と予測類似度は関連が深いと考えられる。表 3 (b) では、類似度ラベルが低いにも関わらず、二つのテキストで同じ一語が生成されており、予測類似度が高くなっていることが確認された。これは、テキストを一語に要約するのが難しい事例では、的確な一語を生成できず、それに伴いその中間表現である条件付きテキスト埋め込みも適切に作用していないことが原因であると考えられる。

図 3 に $\text{PonTE}_{\text{Llama-3-8B-Inst}}$ によるテキスト埋め込みを t-SNE によって二次元に射影した結果を示す。図中の点は、条件に基づきテキスト埋め込みを二次元に射影したベクトルを示し、生成された一語を付与している。表 3 の二つの事例を色で強調してい

る。図 3 から、同じテキストであっても条件が異なれば、離れた位置にあることが確認でき、テキストの表層情報より条件に基づきテキストを埋め込んでいることがわかる。また、生成された一語を見ると、数量に関連する点は右側、生物は下側に集まっているなど、類似した意味の語は近い位置にあることが確認できる。人間の持つ語の意味の近さを表現したテキスト埋め込み空間が得られており、PonTE の有用性が確認できる。

4 実験: テキストクラスタリング

条件付きテキスト埋め込みのもう一つの有力な応用先のテキストクラスタリングで実験を行った。

4.1 実験設定

条件付きテキスト埋め込みを用いることで、様々な側面を条件として埋め込み表現を生成することが可能である。その柔軟性を評価するため、以下の三つのデータセットを用いて複数の側面からテキストクラスタリングを行った。商品レビューのデータセットである Amazon reviews corpus はレビューに商品カテゴリ (Amazon-C) とレーティング (Amazon-R) のラベル、科学系の質問応答データセットである ScienceQA (SciQA) は質問文にトピックのラベル、Tweet emotion intensity dataset (Tweet Emotion) は X (旧 Twitter) の投稿に感情のラベルが付与されており、それぞれラベルごとにテキストクラスタリングを行う。クラスタリングには K-means を用い、クラスタ数にはデータセットごとにラベルの種類数を与えている。シード値を変えて五回実施し、評価指標を V-measure として各評価値の平均をスコアとする。

PonTE では、Mistral 7B と Llama-3 8B のそれぞれベースモデルと指示学習されたモデルの計四つのモデルを用いた。プロンプトテンプレートは、C-STS でスピアマン順位相関係数が最も高いものを用い、プロンプトテンプレートに挿入する条件は検証セットで V-measure の最も高いものを用いた[4)]。比較手法として、C-STS の実験と同じ GTE と E5、条件を与えないテキスト埋め込みとして SimCSE と PromptEOL を導入した。

4.2 実験結果

実験結果を表 4 に示す。PonTE はすべてのデータセットで他の教師なし手法の性能を超え、C-STS の実験と同様に $\text{PonTE}_{\text{Llama-3-8B-Inst}}$ が全体的に高いスコアを示した。PonTE は、トピックのような全体的な意味でクラスタリングするときであっても、明示的に側面を指定することで性能が改善することを示唆している。

手法	Amazon -C	Amazon -R	SciQA	Tweet Emotion
$\text{sup-SimCSE}_{\text{large}}$	19.5	22.4	65.5	29.4
$\text{GTE}_{\text{Qwen2-7B-Inst}}$	**38.3**	36.8	73.9	36.8
$\text{E5}_{\text{Mistral-7B-Inst}}$	37.4	**37.6**	**74.0**	**41.3**
$\text{unsup-SimCSE}_{\text{large}}$	16.7	4.2	63.8	23.4
$\text{PromptEOL}_{\text{Mistral-7B}}$	8.6	27.2	66.0	6.5
$\text{PromptEOL}_{\text{Mistral-7B-Inst}}$	6.1	27.4	59.4	22.7
$\text{PromptEOL}_{\text{Llama-3-8B}}$	9.9	20.4	66.7	9.5
$\text{PromptEOL}_{\text{Llama-3-8B-Inst}}$	9.4	30.8	65.1	31.7
$\text{PonTE}_{\text{Mistral-7B}}$	27.7	27.7	**74.5**	18.1
$\text{PonTE}_{\text{Mistral-7B-Inst}}$	25.3	31.7	68.0	43.8
$\text{PonTE}_{\text{Llama-3-8B}}$	**30.9**	23.8	74.1	24.0
$\text{PonTE}_{\text{Llama-3-8B-Inst}}$	30.5	**34.1**	73.0	**45.9**

表 4 テキストクラスタリングの実験結果。上段が教師あり学習の手法、下段が教師なし学習の手法である。

また、PonTE は教師あり手法と比較しても競争力のある高い性能を示している。特に、$\text{PonTE}_{\text{Mistral-7B-Inst}}$ や $\text{PonTE}_{\text{Llama-3-8B-Inst}}$ は、教師あり SimCSE よりすべてのデータセットで上回るスコアを示した。また、GTE や E5 はクラスタリング対象と類似したデータセットで訓練しているにも関わらず、PonTE はいくつかのデータセットで GTE や E5 を上回るスコアとなった。PonTE は、訓練せずとも有用な条件付きテキスト埋め込みを生成できるため、GTE や E5 がサポートしていない分野や言語であっても容易に応用できる可能性が見込まれる。

5 まとめ

本研究では、訓練不要な条件付きテキスト埋め込み手法 PonTE を提案した。PonTE は、強力な LLM によるテキストエンコーダと条件付き一語制約プロンプトを用いることで、高品質な条件付きテキスト埋め込みを生成できることを実験的に示した。C-STS とテキストクラスタリングの実験では、PonTE は既存の教師なし手法の性能を超え、教師あり手法の性能と同等レベルの性能を達成することを示した。今後の展望として、PonTE の他の応用先や、PonTE が他の分野や言語に適用可能であるか検証したい。

4) 条件の候補は Appendix B に示す

参考文献

[1] Eneko Agirre, Daniel Cer, Mona Diab, and Aitor Gonzalez-Agirre. SemEval-2012 task 6: A pilot on semantic textual similarity. In **Proceedings of *SEM-SemEval 2012**, pp. 385–393, 2012.

[2] Marco Marelli, Stefano Menini, Marco Baroni, Luisa Bentivogli, Raffaella Bernardi, and Roberto Zamparelli. A SICK cure for the evaluation of compositional distributional semantic models. In **Proceedings of LREC 2014**, pp. 216–223, 2014.

[3] Daniel Cer, Mona Diab, Eneko Agirre, Iñigo Lopez-Gazpio, and Lucia Specia. SemEval-2017 task 1: Semantic textual similarity multilingual and crosslingual focused evaluation. In **Proceedings of SemEval 2017**, pp. 1–14, 2017.

[4] Alexis Conneau, Douwe Kiela, Holger Schwenk, Loïc Barrault, and Antoine Bordes. Supervised learning of universal sentence representations from natural language inference data. In **Proceedings of EMNLP 2017**, pp. 670–680, 2017.

[5] Daniel Cer, Yinfei Yang, Sheng-yi Kong, Nan Hua, Nicole Limtiaco, Rhomni St. John, Noah Constant, Mario Guajardo-Cespedes, Steve Yuan, Chris Tar, Brian Strope, and Ray Kurzweil. Universal sentence encoder for English. In **Proceedings of EMNLP 2018**, pp. 169–174, 2018.

[6] Nils Reimers and Iryna Gurevych. Sentence-BERT: Sentence embeddings using Siamese BERT-networks. In **Proceedings of EMNLP-IJCNLP 2019**, pp. 3982–3992, 2019.

[7] Tianyu Gao, Xingcheng Yao, and Danqi Chen. SimCSE: Simple contrastive learning of sentence embeddings. In **Proceedings of EMNLP 2021**, pp. 6894–6910, 2021.

[8] Ameet Deshpande, Carlos Jimenez, Howard Chen, Vishvak Murahari, Victoria Graf, Tanmay Rajpurohit, Ashwin Kalyan, Danqi Chen, and Karthik Narasimhan. C-STS: Conditional semantic textual similarity. In **Proceedings of EMNLP 2023**, pp. 5669–5690, 2023.

[9] Young Yoo, Jii Cha, Changhyeon Kim, and Taeuk Kim. Hyper-CL: Conditioning sentence representations with hypernetworks. In **Proceedings of ACL 2024**, pp. 700–711, 2024.

[10] 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 ACL 2023**, pp. 1102–1121, 2023.

[11] Zehan Li, Xin Zhang, Yanzhao Zhang, Dingkun Long, Pengjun Xie, and Meishan Zhang. Towards general text embeddings with multi-stage contrastive learning. arXiv preprint: 2308.03281, 2023.

[12] Liang Wang, Nan Yang, Xiaolong Huang, Linjun Yang, Rangan Majumder, and Furu Wei. Improving text embeddings with large language models. In **Proceedings of ACL 2024**, pp. 11897–11916, 2024.

[13] Ting Jiang, Shaohan Huang, Zhongzhi Luan, Deqing Wang, and Fuzhen Zhuang. Scaling sentence embeddings with large language models. arXiv preprint: 2307.16645, 2023.

[14] Albert Q. Jiang, Alexandre Sablayrolles, Arthur Mensch, Chris Bamford, Devendra Singh Chaplot, Diego de las Casas, Florian Bressand, Gianna Lengyel, Guillaume Lample, Lucile Saulnier, Lélio Renard Lavaud, Marie-Anne Lachaux, Pierre Stock, Teven Le Scao, Thibaut Lavril, Thomas Wang, Timothée Lacroix, and William El Sayed. Mistral 7b. arXiv preprint: 2310.06825, 2023.

[15] Hugo Touvron, Thibaut Lavril, Gautier Izacard, Xavier Martinet, Marie-Anne Lachaux, Timothée Lacroix, Baptiste Rozière, Naman Goyal, Eric Hambro, Faisal Azhar, Aurelien Rodriguez, Armand Joulin, Edouard Grave, and Guillaume Lample. LLaMA: Open and efficient foundation language models. arXiv preprint: 2302.13971, 2023.

[16] An Yang, Baosong Yang, Binyuan Hui, Bo Zheng, Bowen Yu, Chang Zhou, Chengpeng Li, Chengyuan Li, Dayiheng Liu, Fei Huang, Guanting Dong, Haoran Wei, Huan Lin, Jialong Tang, Jialin Wang, Jian Yang, Jianhong Tu, Jianwei Zhang, Jianxin Ma, Jianxin Yang, Jin Xu, Jingren Zhou, Jinze Bai, Jinzheng He, Junyang Lin, Kai Dang, Keming Lu, Keqin Chen, Kexin Yang, Mei Li, Mingfeng Xue, Na Ni, Pei Zhang, Peng Wang, Ru Peng, Rui Men, Ruize Gao, Runji Lin, Shijie Wang, Shuai Bai, Sinan Tan, Tianhang Zhu, Tianhao Li, Tianyu Liu, Wenbin Ge, Xiaodong Deng, Xiaohuan Zhou, Xingzhang Ren, Xinyu Zhang, Xipin Wei, Xuancheng Ren, Xuejing Liu, Yang Fan, Yang Yao, Yichang Zhang, Yu Wan, Yunfei Chu, Yuqiong Liu, Zeyu Cui, Zhenru Zhang, Zhifang Guo, and Zhihao Fan. Qwen2 technical report. arXiv preprint: 2407.10671, 2024.

[17] 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 preprint: 2212.03533, 2024.

A　プロンプトテンプレートの影響

表 5 に $\text{PonTE}_{\text{Llama-3-8B-Inst}}$ の検証セットにおけるプロンプトテンプレートごとの C-STS の実験結果を示す。(1)-(6) は PromptEOL で使用されたものを発展したプロンプト、(7)-(12) は指示形式にしたプロンプトである。表 5 から「in one word」がスコアに大きく影響していることがわかる。また、(7)-(12) は (1)-(6) より全体的にスコアが高く、指示形式にすることで、プロンプトの指示をより反映し、性能が高くなっていると考えられる。「in terms of」と「with respect to」による差は比較的小さかった。

プロンプトテンプレート	r_s	r_p
{p} = This text: "{text}" means		
(1) {p} **in terms of {condition}**: "	18.8	17.1
(2) {p} **with respect to {condition}**: "	18.0	17.0
(3) {p} in one word **in terms of {condition}**: "	28.2	25.2
(4) {p} in one word with respect to {condition}: "	25.1	21.7
(5) {p} **in terms of {condition}** in one word: "	28.1	24.7
(6) {p} with respect to {condition} in one word: "	25.4	22.3
{p} = Express this text "{text}"		
(7) {p} **in terms of {condition}**: "	19.8	18.2
(8) {p} with respect to {condition}: "	19.1	17.6
(9) {p} in one word **in terms of {condition}**: "	**37.3**	**34.8**
(10) {p} in one word with respect to {condition}: "	36.4	33.9
(11) {p} **in terms of {condition}** in one word: "	33.1	30.4
(12) {p} with respect to {condition} in one word: "	30.4	27.9

表 5　プロンプトテンプレートごとの C-STS の実験結果

B　条件の影響

表 6 に各データセットの検証セットにおける条件ごとのテキストクラスタリングの実験結果を示す。Amazon-C や SciQA では、「name」や「product」、「question」を加えた条件のスコアが高く、Amazon-R でも「star」や「rating」だけよりも「star rating」を入れたもののスコアが高くなっている。これは条件を具体的にすることで LLM がプロンプトの意図をより反映した出力ができ、スコアが高くなったと考えられる。

C　データセットの統計

C-STS の実験では、Deshpande ら [8] による C-STS データセット[5]を用いた。テキストクラスタリングの実験では、Amazon reviews corpus[6]、ScienceQA[7]、Tweet emotion intensity dataset[8]の三つのデータセットを用いた。データセットの統計を表 7 に示す。

	条件	V-measure
Amazon -C	(a) the category	22.8
	(b) the product category	26.5
	(c) the category name	21.6
	(d) the product category name	29.4
	(e) the name of the category	22.0
	(f) the name of the product category	**30.5**
Amazon -R	(g) the rating	30.4
	(h) the star	26.2
	(i) the star rating	**34.1**
	(j) the five-level rating	21.3
	(k) the five-level star rating	29.3
	(l) the emotion	33.4
SciQA	(m) the category	70.3
	(n) the question category	74.2
	(o) the name of the category	74.1
	(p) the name of the question category	**75.4**
Tweet Emotion	(q) the emotion	**45.9**
	(r) the feeling	44.7
	(s) the sentiment	43.6

表 6　条件ごとのテキストクラスタリングの実験結果

データセット	ラベル数	検証セット	テストセット
C-STS	-	2,840	4,732
Amazon-C	31	5,000	5,000
Amazon-R	5	5,000	5,000
SciQA	25	4,241	4,241
Tweet Emotion	4	374	1,421

表 7　実験で使用したデータセットの統計

D　PonTE で用いたモデルのリンク

PonTE は六つの LLM を用いて実験を行った。具体的には、Mistral 7B [14] のベースモデル[9]と指示学習されたモデル[10]、Llama-3 8B [15] のベースモデル[11]と指示学習されたモデル[12]、Llama-3 70B のベースモデル[13]と指示学習されたモデル[14]を用いた。

5）princeton-nlp/c-sts
6）mexwell/amazon-reviews-multi
7）derek-thomas/ScienceQA
8）cardiffnlp/tweet_eval
9）mistralai/Mistral-7B-v0.3
10）mistralai/Mistral-7B-Instruct-v0.3
11）meta-llama/Meta-Llama-3-8B
12）meta-llama/Meta-Llama-3-8B-Instruct
13）meta-llama/Meta-Llama-3-70B
14）meta-llama/Meta-Llama-3-70B-Instruct