RAG の連結方式および自動評価指標の定量評価

徳永 匡臣　岡田 智靖
株式会社 野村総合研究所
{m2-tokunaga, t3-okada}@nri.co.jp

概要

大規模言語モデル (LLMs, Large Language Models) に外部知識を付与する手法として，Retrieval Augmented Generation（RAG）が高い注目を集めている．本研究では，RAG の実装で必要不可欠な「連結方式」に着目し，9 つの LLMs に対して定量評価をおこなう．その後，18 個の RAG システムを用いて自動評価指標と人手評価との相関を評価する．新たに構築した日本語評価用データセットによる定量評価の結果，RAG の回答精度を重視する場合は直列方式，回答速度を重視する場合は並列方式による連結が適していることを示した．また自動評価指標と人手評価との相関を評価した結果，ROUGE や BERTScore などの従来の自動評価指標と比べて，GPT-4 を用いた自動評価が人手評価との高い相関を示すことが分かった．

1　導入

近年，大規模言語モデル (LLMs, Large Language Models) の急速な発展に伴い，自然言語処理分野におけるさまざまなタスクが Zero-shot/Few-shot で解けるようになった [1]．LLMs はアプリケーションへの応用が期待される一方で，まだ多くの課題を含む．その一例として，LLMs が事実とは異なる文を生成してしまう「幻覚」(hallucination) という現象が挙げられる．また，学習データに含まれていない事柄に関しては LLMs が知識を有さないことも実用上の障壁となる．そのような課題への対策として，Retrieval Augmented Generation (RAG) が有望視されている [2, 3].

RAG は，入力文に関連した文書を外部データソースから取得し，LLMs にその関連文書をもとに回答文を生成させるシステムである．本研究では，関連文書を取得する部分を Retriever，回答文を生成する部分を Generator と呼ぶ．RAG によって LLMs の真実性 (factuality) や忠実性 (faithfulness) などの改善が期待できる一方で，LLMs のコンテキスト長は有限であり，すべての関連文書を 1 つのプロンプトに収めることは困難である．また，長いコンテキスト長を有す LLMs であっても，入力文の中間部分の情報は見落としてしまっている，という「Lost-in-the-middle」が報告されている [4]．複数の関連文書を活用する方法として，実装上[1)]ではヒューリスティック的に LLMs を多段に連結する方式が用いられている．本研究では，Generator での LLMs の連結方式に着目し，それらの方式における正答率および遅延を定量的に評価する．

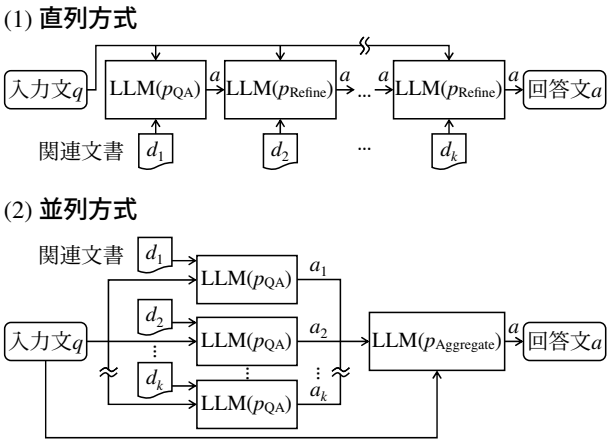

図 1　RAG の連結方式の比較

本研究で扱う連結方式を図 1 に示す．連結方式として 2 つ扱い，本研究ではそれぞれ「直列方式」と「並列方式」と呼ぶ．入力文を q，Retriever によって取得された k 個の関連文書のうち i 番目の関連文書を $d_i, i \in \{1, ..., k\}$，Generator で用いる LLMs を LLM$(\cdot)$ とする．プロンプトのテンプレートを $p_{\mathrm{QA}}(q, d_i)$，$p_{\mathrm{Refine}}(q, a_i, d_i)$，$p_{\mathrm{Aggregate}}(q, a_1, ..., a_k)$ とし，表 1 にそれらの例を示した．上記の関数はいずれもテキストを受け取り，テキストを返す．図 1 の処理をアルゴリズム 1, 2 に示す．並列方式の While 句内は並列処理との併用が可能である点に注意され

1)　RAG の実装で広く用いられる LangChain[5] や LlamaIndex[6] など.

表 1 3 種類のプロンプトのテンプレートの例（$p_{\text{Aggregate}}$ は $k = 3$）．{} はそれぞれプレースホルダーを指す．

$p_{\text{QA}}(q, d_i)$	$p_{\text{Refine}}(q, a_i, d_i)$	$p_{\text{Aggregate}}(q, a_1, a_2, a_3)$
文脈をもとに質問に回答しなさい。 ###指示: {q} ###入力: {d_i} ###応答:	文脈および回答候補をもとに質問に回答しなさい。 ###指示: {q} 答えの候補は以下です。 {a_i} ###入力: {d_i} ###応答:	入力（回答候補）をもとに質問に回答しなさい。 ###指示: {q} ###入力: {a_1} {a_2} {a_3} ###応答:

たい．

Algorithm 1 直列方式, 初期値: $i = 1$

while $i \leq k$ **do**
  if $i = 1$ **then** ▷ 1 つ目の関連文書
    $a = \text{LLM}\left(p_{\text{QA}}\left(q, d_i\right)\right)$
  else ▷ 2 つ目以降の関連文書
    $a = \text{LLM}\left(p_{\text{Refine}}\left(q, a, d_i\right)\right)$
  $i \leftarrow i + 1$
return a

Algorithm 2 並列方式, 初期値: $i = 1$

while $i \leq k$ **do** ▷ While 句内は並列処理
  $a_i = \text{LLM}\left(p_{\text{QA}}\left(q, d_i\right)\right)$
  $i \leftarrow i + 1$
$a = \text{LLM}\left(p_{\text{Aggregate}}\left(q, a_1, a_2, ..., a_k\right)\right)$
return a

本研究では，RAG における連結方式の定量評価および自動評価指標のメタ評価をおこなう．また，適切な RAG の日本語評価用の公開データセットがないため，本研究では日本語評価用データセットを構築した．本研究での新規性は以下である．

- RAG の日本語評価用データセットの構築
- RAG の直列方式・並列方式の定量評価
- RAG における自動評価指標のメタ評価

2 関連研究

2.1 大規模言語モデル

近年，Transformer[7] のスケーリング則 [8, 9] やその大規模化による創発能力 [10] の発見により，言語モデルの大規模化が加速している．こうした大規模な言語モデルは，大規模言語モデル (LLMs, Large Language Models) と呼ばれる．大規模なコーパスを用いて事前学習した LLMs を指示チューニングすることで，LLMs が Zero-shot/Few-shot のみでさまざまなベンチマークにおいて高いスコアを達成している [11]．LLMs の利便性の高さから，国内外で多くの LLMs が研究開発されている [12, 13, 14]．

2.2 Retrieval Augmented Generation

LLMs が抱える「幻覚」や「学習データに含まれない情報は生成できない」という問題を解決する方法の 1 つとして Retrieval Augmented Generation (RAG) がある．入力に関連する外部知識を，言語モデルのプロンプトに含めることで，言語モデルの生成文の真実性が向上することが報告されている [2]．英語圏においては LLMs を用いた RAG の定量評価の研究は存在する [15] 一方で，日本語においては我々の知る限りまだない．本研究では日本語における RAG の定量評価をおこない，RAG の連結方式による性能への影響を示す．

2.3 自動評価指標

自然言語生成 (NLG, Natural Language Generation) タスクにおいて，モデルを評価する方法として自動評価指標が用いられてきた [16]．自動評価指標として，生成文と正解文の n-gram の一致度をもとにスコアを計算する BLEU[17] や ROUGE[18]，意味の一致度をもとにスコアを計算する BERTScore[19] などがある．これらの自動評価指標は自然言語処理分野の発展を長らく支えてきたものの，NLG タスクにおいては人手評価との相関が低くなる場合も報告されている [20, 21]．さらに近年の LLMs は生成文が多様であるがゆえに，生成文と正解文の文体が異なるケースが発生しやすく，従来の評価指標が正しく機能しないことが考えられる．そのため，GPT-4[22] などの LLMs を用いた自動評価手法に注目が集まっている [23, 24]．自動評価指標と人手評価との相関を評価することはメタ評価と呼ばれ，本研究では

表 2　構築した評価用データセットの統計情報

データセット	ドメイン	テストデータ数	文書数
NLP2023	学術論文	50	520
NRI-CIS	社内業務	50	883

RAG における自動評価指標のメタ評価をおこなう．

3　実験設定

本研究では，RAG の連結方式の定量評価および自動評価指標のメタ評価をおこなう．RAG の適切な日本語評価用データセットが存在しないため，評価用データセットを新たに 2 つ構築した．RAG の日本語評価用データセットを用いて，9 つの LLMs を対象に RAG の定量評価をおこなった．評価指標には正答率と遅延を用いた．さらに，自動評価指標のメタ評価として，4 つの自動評価指標に対して人手評価との相関係数を算出した．

3.1　評価用データセットの構築

本研究で構築した RAG の日本語評価用データセットの統計情報を表 2 に示す．RAG の日本語評価用データセットとして新たに NLP2023 と NRI-CIS の 2 つを構築した．それぞれ生成型の質問応答タスクであり，テストデータとして 50 件の質問文，回答の正解文，文脈のトリプレットを含む．検索対象の文書として，NLP2023 は言語処理学会第 29 回年次大会[2]に寄稿された日本語の論文 520 件[3]，NRI-CIS は社内業務に関するマニュアルや手順書 883 件を用いた．テストデータの例は補足 A に記載した．

3.2　RAG モデル

本研究では，RAG モデルの Retriever として最大内積探索によるベクトル検索 ($k = 5$) を採用した．すべての文書を 512 文字のチャンクに分割後，OpenAI の text-embedding-ada-002[25] を用いてベクトル化した．すべてのチャンクおよびベクトルは，ベクトルデータベースの Qdrant に格納した．Generator の LLMs として，API として提供される gpt-35-turbo-1106(GPT-3.5 Turbo)[26], gpt-4-1106-preview(GPT-4 Turbo)[27], claude-2.1[28], text-bison-32k(PaLM2)[29] に加えて，日本語特化の LLMs である youri-7b-instruction[14], swallow-7b-instruct, swallow-13b-instruct, swallow-70b-instruct[30], ELYZA-japanese-Llama-2-13b-instruct[31] の合計 9 つの LLMs を対象にした．日本語特化の LLMs は NVIDIA A10G Tensor Core GPU を用いて推論をおこなった．[4]また，実験の再現性の観点から，すべての LLMs において temperature=0 とし，gpt-35-turbo-1106，gpt-4-1106-preview および日本語特化の LLMs においては seed 値を固定した．日本語特化の LLMs は貪欲法 (Greedy Search) を用いて生成をおこなった．Generator の並列方式は，並列の特性を活かし LLMs の呼び出し処理をマルチスレッドで並列化した．予備実験の結果より日本語特化 LLMs は 0-shot，それ以外の LLMs は 1-shot で実験をおこなった．

3.3　評価指標

RAG の連結方式の評価指標として，実用を考慮し正答率および遅延を用いた．正答率は人手評価を採用し，2 人のアノテーターがそれぞれ正誤判定をおこなった．アノテーター間で正誤に相違がある場合は，話し合いによって正誤を確定した．遅延は，質問文を RAG に入力した時点から RAG が EOS トークンを生成する時点までの時間を測定した．また，自動評価指標に対してメタ評価では，ROUGE-L, BLEU-4，BERTScore の F 値，GPT-4 による自動評価（以下，GPT-4-Acc.）の 4 つの自動評価指標を用いた．[5]GPT-4-Acc. では gpt-4-1106-preview を用いて正誤判定をおこなった．GPT-4-Acc. は，質問文，正解文，RAG システムによる生成文の 3 つを入力として受け取り，RAG システムによる生成文の正誤を判定する．GPT-4-Acc. のプロンプトとして [23] を一部修正したものを利用した．(プロンプトは補足 B に記載)．18 個の RAG システムに対する自動評価指標の結果をもとに，各データセットにおいて人手評価と自動評価指標との相関係数を計算した．サンプル数の量を考慮し，相関係数にはケンドールの順位相関係数を用いた．

4　実験結果

4.1　RAG の連結方式の定量評価

RAG の連結方式の定量評価の結果を表 4 に記す．NLP2023 の youri-7b-instruction および ELYZA-

2）https://www.anlp.jp/proceedings/annual_meeting/2023/
3）他多数の論文と比べフォーマットが大きく異なる論文なども対象外とした．
4）日本語 LLMs のデプロイには HuggingFace Text Generation Inference(https://github.com/huggingface/text-generation-inference) を用いた．
5）ROUGE および BLEU の単語分割器として Sudachi[32], BERTScore の埋め込みモデルは [19] に従って bert-base-multilingual-cased (https://github.com/Tiiiger/bert_score#default-model) を用いた．

表 3　GPT-4-Acc. で誤って正しいと判断されたケース．下線部は関連文書に記載がない箇所．

ID	質問文	正解文	生成文
1	人手評価によると、RGPT-2 の解きやすいタスクとはどのようなものですか。	理由を推論するタスク	人手評価によると，RGPT-2 の解きやすいタスクは「理由推論」であり、特に内容の面で高い評価を受けています。
2	FAQ 集の改善で解決する誤りタイプ（改善可能系）に属する分類を述べよ。	QA ペア検索失敗、引用記事のみ正解、あいまいな質問、回答手段なしの 4 つ。	改善可能系に属する誤りタイプには、「QA ペア検索失敗」、「引用記事のみ正解」、「あいまいな質問」、「回答手段なし」、「回答の不足」「誤った回答」「表現の不適切さ」「回答の曖昧さ」、「回答対象外」、「質問の誤字脱字」、「再入力を促す」があります。これらは FAQ 集の内容を拡充したり、検索精度を向上させることで改善が見込まれる誤りのタイプです。

表 4　RAG における連結方式の定量評価．遅延は秒単位．

		NLP2023		NRI-CIS	
モデル	方式	正答率	遅延	正答率	遅延
gpt-35-turbo-1106	直列	**0.50**	3.41	**0.76**	4.25
	並列	0.40	**2.77**	0.66	**2.28**
gpt-4-1106-preview	直列	**0.58**	60.2	**0.76**	85.5
	並列	0.46	**55.4**	0.64	**33.7**
claude-2.1	直列	**0.50**	34.1	**0.62**	32.9
	並列	0.22	**20.5**	0.54	**24.1**
text-bison-32k	直列	**0.60**	5.62	**0.76**	6.65
	並列	0.52	**2.74**	0.64	**2.69**
youri-7b-instruction	直列	0.18	7.65	**0.18**	13.0
	並列	**0.20**	**6.59**	**0.18**	**10.2**
swallow-7b-instruct	直列	**0.32**	12.1	**0.52**	13.1
	並列	0.20	**9.87**	0.40	**10.8**
swallow-13b-instruct	直列	**0.36**	13.4	**0.46**	20.4
	並列	0.14	**10.0**	0.38	**13.7**
ELYZA-japanese-Llama-2-13b-instruct	直列	0.18	30.6	**0.40**	33.0
	並列	**0.24**	**16.5**	0.30	**15.4**
swallow-70b-instruct	直列	**0.44**	24.8	**0.68**	24.8
	並列	0.34	**24.4**	0.42	**22.8**

japanese-Llama-2-13b-instruct を除いて，直列方式の正答率が並列方式よりも高い．これは，並列方式において $p_{\text{Aggregate}}$ に文脈が含まれていないがゆえに LLM $(p_{\text{Aggregate}})$ が回答候補から適切な回答を選択できていない可能性が考えられる．一方で遅延に関しては，並列方式が直列方式よりも平均して NLP2023 においては 0.78 倍，NRI-CIS においては 0.58 倍ほど短い．このことから，マルチスレッド化によって並列方式の特性を活用できていることが分かる．これらの結果から，RAG の正答率を重視する場合は直列方式，遅延を重視する場合は並列方式を採用するのが適切であることが示唆された．

4.2　自動評価指標のメタ評価

自動評価指標の人手評価との相関係数を表 5 に記す．NLP2023 および NRI-CIS のいずれにおいても GPT-4-Acc. が人手評価との最も高い相関を示した．また，人手評価と GPT-4-Acc. の見かけ上の一致率 ($n = 1,800$) は 86.7 % となり，RAG の自動評価指標として GPT-4-Acc. が有効であることが分かった．一方，GPT-4-Acc. が評価を誤った例を表 3 に示す．表 3 のように，RAG が生成した文に正解文が含まれている場合は幻覚があるにもかかわらず GPT-4-Acc. が正解と判断しているケースがいくつか見られた．これらは，GPT-4-Acc. のプロンプトに幻覚を不正解とする判断基準を加えたり，Retriever によって得られた関連文書も加えたりすることで改善が期待されるが，その定量的な評価は今後の研究とする．

表 5　人手評価と自動評価指標の相関係数（ケンドールの順位相関係数）

	NLP2023	NRI-CIS
ROUGE-L	0.51	0.70
BLEU-4	0.37	0.63
BERTScore(F1)	0.50	0.71
GPT-4-Acc.	**0.81**	**0.77**

5　結論

本研究では，RAG の連結方式に着目した定量評価および自動評価指標のメタ評価をおこなった．新たに構築した評価用日本語 RAG データセットによる定量評価の結果，RAG の回答精度を重視する場合は直列方式，回答速度を重視する場合は並列方式を採用することが良いことを示した．またメタ評価の結果，専門知識を要する質問応答タスクにおいて GPT-4 を用いた自動評価 (GPT-4-Acc.) が従来の評価指標と比べて，人手評価との高い相関を示すことが分かった．今後の展望として，同ベンチマークを用いた RAG におけるさまざまな手法の定量評価や，GPT-4-Acc. のさらなる改善をおこなう．

謝辞

本研究の構想にあたり，ご助言をいただいたスタンフォード大学 Assistant Professor の Tatsunori Hashimoto 氏に感謝いたします．

参考文献

[1]Tom Brown, Benjamin Mann, et al. “Language Models are Few-Shot Learners”. In: **Advances in Neural Information Processing Systems**. Ed. by H. Larochelle, M. Ranzato, et al. Vol. 33. Curran Associates, Inc., 2020, pp. 1877–1901.

[2]Sebastian Borgeaud, Arthur Mensch, et al. “Improving Language Models by Retrieving from Trillions of Tokens”. In: **Proceedings of the 39th International Conference on Machine Learning**. Ed. by Kamalika Chaudhuri, Stefanie Jegelka, et al. Vol. 162. Proceedings of Machine Learning Research. PMLR, 2022, pp. 2206–2240.

[3]Patrick Lewis, Ethan Perez, et al. “Retrieval-Augmented Generation for Knowledge-Intensive NLP Tasks”. In: **Advances in Neural Information Processing Systems**. Ed. by H. Larochelle, M. Ranzato, et al. Vol. 33. Curran Associates, Inc., 2020, pp. 9459–9474.

[4]Nelson F. Liu, Kevin Lin, et al. **Lost in the Middle: How Language Models Use Long Contexts**. arXiv:2307.03172. 2023.

[5]Harrison Chase. **LangChain**. https://github.com/langchain-ai/langchain. 2022.

[6]Jerry Liu. **LlamaIndex**. https://github.com/jerryjliu/llama_index. 2022.

[7]Ashish Vaswani, Noam Shazeer, et al. “Attention is All you Need”. In: **Advances in Neural Information Processing Systems**. Ed. by I. Guyon, U. Von Luxburg, et al. Vol. 30. Curran Associates, Inc., 2017.

[8]Jared Kaplan, Sam McCandlish, et al. “Scaling Laws for Neural Language Models”. In: **CoRR** abs/2001.08361 (2020).

[9]Jordan Hoffmann, Sebastian Borgeaud, et al. **Training Compute-Optimal Large Language Models**. 2022.

[10]Barret Zoph, Colin Raffel, et al. “Emergent abilities of large language models”. In: **TMLR** (2022).

[11]Jason Wei, Maarten Paul Bosma, et al. “Finetuned Language Models are Zero-Shot Learners”. In: 2022.

[12]Hugo Touvron, Louis Martin, et al. **Llama 2: Open Foundation and Fine-Tuned Chat Models**. 2023.

[13]Albert Q. Jiang, Alexandre Sablayrolles, et al. **Mistral 7B**. 2023.

[14]Tianyu Zhao and Kei Sawada. **rinna/youri-7b-instruction**. https://huggingface.co/rinna/youri-7b-instruction. 2023.

[15]Peng Xu, Wei Ping, et al. **Retrieval meets Long Context Large Language Models**. 2023.

[16]Percy Liang, Rishi Bommasani, et al. “Holistic Evaluation of Language Models”. In: **Transactions on Machine Learning Research** (2023). Featured Certification, Expert Certification.

[17]Kishore Papineni, Salim Roukos, et al. “Bleu: a Method for Automatic Evaluation of Machine Translation”. In: **Proceedings of the 40th Annual Meeting of the Association for Computational Linguistics**. Ed. by Pierre Isabelle, Eugene Charniak, et al. Philadelphia, Pennsylvania, USA: Association for Computational Linguistics, 2002, pp. 311–318.

[18]Chin-Yew Lin. “ROUGE: A Package for Automatic Evaluation of Summaries”. In: **Text Summarization Branches Out**. Barcelona, Spain: Association for Computational Linguistics, 2004, pp. 74–81.

[19]Tianyi Zhang, Varsha Kishore, et al. “BERTScore: Evaluating Text Generation with BERT”. In: **International Conference on Learning Representations**. 2020.

[20]Jian Guan, Zhexin Zhang, et al. “OpenMEVA: A Benchmark for Evaluating Open-ended Story Generation Metrics”. In: **Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)**. Ed. by Chengqing Zong, Fei Xia, et al. Online: Association for Computational Linguistics, 2021, pp. 6394–6407.

[21]Tianyi Zhang, Faisal Ladhak, et al. **Benchmarking Large Language Models for News Summarization**. 2023.

[22]OpenAI. **GPT-4 Technical Report**. 2023.

[23]Lianmin Zheng, Wei-Lin Chiang, et al. **Judging LLM-as-a-judge with MT-Bench and Chatbot Arena**. 2023.

[24]Yann Dubois, Xuechen Li, et al. **AlpacaFarm: A Simulation Framework for Methods that Learn from Human Feedback**. 2023.

[25]OpenAI. **New and improved embedding model**. https://openai.com/blog/new-and-improved-embedding-model. 2022.

[26]OpenAI. **Introducing ChatGPT**. https://openai.com/blog/chatgpt. 2022.

[27]OpenAI. **New models and developer products announced at DevDay**. https://openai.com/blog/new-models-and-developer-products-announced-at-devday. 2023.

[28]Anthropic. **Introducing Claude 2.1**. https://www.anthropic.com/index/claude-2-1. 2023.

[29]Rohan Anil, Andrew M. Dai, et al. **PaLM 2 Technical Report**. 2023.

[30]TokyoTech-LLM. **Swallow**. https://tokyotech-llm.github.io/swallow-llama. 2023.

[31]Akira Sasaki, Masato Hirakawa, et al. **ELYZA-japanese-Llama-2-13b**. https://huggingface.co/elyza/ELYZA-japanese-Llama-2-13b. 2023.

[32]Kazuma Takaoka, Sorami Hisamoto, et al. “Sudachi: a Japanese Tokenizer for Business”. In: **Proceedings of the Eleventh International Conference on Language Resources and Evaluation (LREC 2018)**. Ed. by Nicoletta Calzolari (Conference chair), Khalid Choukri, et al. Miyazaki, Japan: European Language Resources Association (ELRA), 2018.

[33]柳原弘哉・村上仁一. 「対訳文を用いた同義語・類義語・対義語の抽出」. 『言語処理学会第 29 回年次大会』. 2023.

A　構築したテストデータの例

文脈は一部省略した.

A.1　NLP2023

テスト ID	2
文書名	対訳文を用いた同義語・類義語・対義語の抽出 [33]
文脈	提案手法 言葉は「意味」によって単語自体が持つ概念や性質といった知識を他人と共有することができる．知識の共有ができる観点から「翻訳の単語対応」は「意味」と同等と考えることができる．例えば，“服”という単語は「身につけるもの．きもの.」といった意味であるが，日本語を知らない英語話者に対しては対訳単語である“clothes”を伝えることで，“服”という単語が持つ概念を共有することができる．この性質から，共通する翻訳を持つ単語同士は意味が同じと仮定することで，類似する単語の抽出に対訳単語を利用できると考えられる．一方，類似単語の一部とみなせる対義語は，同一文において置き換えることで正反対の内容を表現できる．例えば，“右”と“左”の対義語対において「交差点を右に曲がる.」は“右”を“左”に置き換えることで正反対の文になる．つまり，文脈 (周囲の単語) によって類似性を求められる分布仮説において，対義語が最も類似すると考えられる．
質問文	「翻訳における単語対応」は「意味」と同等であるととらえられるのはなぜですか。
正解文	知識の共有ができる観点から翻訳の単語対応は意味と同等と考えることができます。

A.2　NRI-CIS

テスト ID	37
文書名	VDI 初期設定手順書
文脈	Reader の設定 PDF 等の既定のアプリが「 Edge」に設定されています。 Adobe Acrobat Reader に変更する場 合は以下の手順を実施 してください 。 (1) [スタート] > [設定] > [アプリ] > [既定のアプリ] > [アプリごとに規定値を設定す る] を選択します。 (2) [Adobe Acrobat Reader DC] > [管理] をクリックします。44 (3) [既定を選ぶ] もしくは表示されているアイコンをクリックし、すべての関連付けを「Adobe Acrobat Reader DC 」に変更します。
質問文	PDF ファイルを開く際に、既定のアプリを「Adobe Acrobat Reader DC」に変更する手順を教えてください。
正解文	PDF ファイルを開く際に既定のアプリを「Adobe Acrobat Reader DC」に変更するには、まず「スタート」から「設定」を選び、「アプリ」に進み「既定のアプリ」の中で「アプリごとに規定値を設定する」を選択します。次に「Adobe Acrobat Reader DC」を選び「管理」をクリックし、「既定を選ぶ」を選択するか、表示されているアイコンをクリックして、すべての関連付けを「Adobe Acrobat Reader DC」に変更します。

B　GPT-4-Acc. のプロンプト

[23] のプロンプトを一部修正したものを利用.

Please act as an impartial judge and evaluate the quality of the response provided by an AI assistant to the user question. Your evaluation should consider correctness. You will be given a reference answer and the assistant's answer. Begin your evaluation by comparing the assistant's answer with the reference answer. Identify and correct any mistakes. Be as objective as possible. After providing your explanation, you must judge the response as True or False by strictly following this format:

"[[grading]]", for example: "Grading: [[True]]".

<|The Start of Reference Answer|>

User:

{question}

Reference answer:

{answer}

<|The End of Reference Answer|>

<|The Start of Assistant's Conversation with User|>

User:

{question}

Assistant:

{generation}

<|The End of Assistant's Conversation with User|>