大規模言語モデルの数値データ説明における例示と補足の効果

江部 正周 [1] 青山 敦 [2]
[1] 慶應義塾大学 SFC 研究所 [2] 慶應義塾大学 環境情報学部
{ebe,aaoyama}@sfc.keio.ac.jp

概要

大規模言語モデル（LLM）は, 質疑応答等の定性的な推論や数学等の定量的な推論が可能である. 一方で, 定性的な推論と定量的な推論を組み合わせることが必要な CSV データのような数値データを, 統計知識に基づいて自然言語で説明できるかは明らかでない. 本研究では, 数値データの統計知識に基づいた説明において, 説明能力を向上させる効果が高くコストの低い手法を, 次の 3 種類の LLM への指示で検証した. すなわち, 1) 例も補足もない指示, 2) 基本的な統計量等の例示, 3) LLM の役割や分析の目的等の補足, である. その結果, 例示よりも補足の方が説明能力を向上させる効果は同等でもコストは低く, 数値データを統計知識に基づいて説明できた.

1 はじめに

人間と人工知能との協働的なデータ分析において, CSV データのような数値データを, 統計知識に基づいて自然言語で説明することは重要である. 人間との自然な応答を可能とする大規模言語モデル（LLM）は, 自然言語での定性的な推論 [1] において飛躍的に発展してきた. さらには, 近年, 初等数学 [2, 3] のみならず, 時系列予測 [4, 5] のような定量的な推論も, LLM の訓練によって可能であることが報告されている. 数値データを自然言語で説明するには, 定量的な推論と定性的な推論の双方が必要だと考えられるが, これらの推論を組み合わせることで, 数値データを統計知識に基づいて説明できるかは明らかでない.

LLM が数値データを自然言語で説明するには, まずデータからテキストを生成できること, 次に意味のある洞察を得られることが必要だと考えられる. データからテキストを生成する課題においては, 主に加工済みのデータからのテキスト生成やデータへのコメント生成を中心に研究されてきた. 例えば, チャート [6] や数値を含むテキスト [7] を説明させる研究では, 数値データを分析した加工済みの情報をテキストに変換できることが報告されている. また, 市場データを対象とした研究 [8] では, 市場を解説するような付加的なコメントを生成できることが報告されている. つまり, データからテキストを生成することは可能であると言える. また, データから意味のある洞察を得る課題では, LLM が自律的に分析や考察を行うことを目的とした研究がされてきた. 例えば, LLM が外部ツールを用いて, データ分析 [9, 10] から科学的な発見 [11, 12, 13] まで, 自律的にデータから洞察を得ることが部分的に可能であることが報告されている. したがって, CSV データのような数値データについても, LLM に数値データを説明する訓練をすることで, データの特徴を人間が理解できるように統計知識に基づいて説明できる可能性がある.

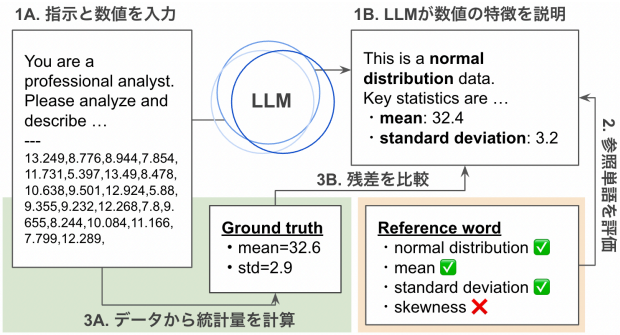

図 1 本研究の概要図

本研究では, LLM による定性的な推論と定量的な推論を組み合わせ, 統計知識に基づいた数値データの説明において, 説明能力を向上させる手法を明らかにすることを目的とした. このとき, LLM の実務での活用を想定し, 1) LLM の訓練コスト（計算コスト等）, 2) LLM の推論コスト（API 利用コスト等）, 3) LLM 利用者の学習コスト（指示に専門知識が必要等）にも着目した. LLM への指示と評価の流れについては, 図 1 の通りである. まず, LLM の訓練手法は, 訓練コストの必要ない LLM の入力（プロンプ

ト）のみでの学習, すなわち, 文脈内学習 [14] とした. 次に, データ分析に関する統計知識が LLM 利用者にとって必要かという観点でプロンプトを条件分けして, 説明能力の検証を行った. 最後に, API 利用コストに関連する LLM への入力文字数に着目して, プロンプトのコストを比較した.

2 実験方法

2.1 実験条件

LLM による数値データの説明において, 統計知識に基づいて数値データを説明する能力が, どのような指示によって向上するかを検証するため, 次に説明する 3 つの LLM への指示を実験条件とした.

Baseline 条件 数値データの説明だけを求めた LLM への指示によって, 説明能力を検証した. データ分析では, 対象のデータにおける平均や分散等の基本的な統計量を確認することが標準的な手続きである. しかし, データ分析の非専門家にとってはそうではない. 一般的な知識を学習している LLM にとって, 統計知識に基いてデータの基本的な統計量を説明することは標準的な手続きではない可能性がある. これを確認するため, 数値データについての網羅的な説明, または重要な点の説明だけを指示する Baseline 条件を設けた.

Example 条件 数値データの説明において, 基本的な統計量等の例を LLM に与えることで, 統計知識に基づいた説明能力を引き出すことができるか検証した. LLM はいくつかの例を指示に加える少数ショット学習によって能力が向上し [14], 事前に訓練していない数学等の課題についても解くことができるようになる [15]. この少数ショット学習を用い, 説明すべき基本的な統計量等の例を与えて説明能力の向上を図る Example 条件を設けた.

Supplement 条件 Example 条件とは異なり, 基本的な統計量等の例を含まない指示方法で, 説明能力を向上させることができるか検証した. その背景として, 基本的な統計量等の例を作成するには, LLM 利用者の統計の専門知識が必要となるため, 学習コストが高いという問題がある. LLM に「優秀な助手」や「先生」等の役割を示すと, 能力が向上すること [16, 17, 18] や, データの関連情報を与えると, 時系列予測の精度が上がることが知られている [19, 20]. これらの知見を踏まえ, 数値データの説明においても, LLM の役割や分析の目的等を補足することで, 例示と同等の効果を得られる可能性がある. このように, 指示に補足することで基本的な統計量を説明できるかを検証する Supplement 条件を設けた.

以上の指示条件と数値データをプロンプトに組み込み, 複数の LLM のモデルにデータについて説明させた. 実験は次に示すように, プロンプト 9 種類, データセット 40 種類, モデル 15 種類の合計 5,400 試行を行った.

2.2 プロンプト

実験条件で示した 3 つの指示について, 指示に続くカンマで区切られた 100 個の数値データを説明するようにプロンプトを作成した. プロンプトの日本語での簡略例を表 1 に示した.

表 1 プロンプト条件と日本語での簡略例

条件名	プロンプトの日本語での簡略例
Baseline	次のデータを**網羅的**に説明してください. 10.254,12.804,10.63,8.283,9.468,8.702,...
Example	次のデータの**平均**と**分散**を説明してください. 8.283,12.804,10.63,9.468,8.702,10.254,...
Supplement	あなたは**データ分析の専門家**です. 次のデータを説明してください. 9.468,10.254,12.804,10.63,8.283,8.702,...

これらの 3 条件に対して 3 種類ずつ, 9 種類のプロンプトを用意した. なお, 言語は英語を用いた. プロンプトの具体例については, 付録 A.1 に示した.

2.3 データセット

1–2 桁の小数点第 3 位までの数値データ（e.g., 3.141）100 個を Python NumPy ライブラリ [21] を用いて生成した. これらは, 一峰性の正規分布, 二峰性の正規分布, 一様分布, 冪乗分布に従うランダムな数値として, 異なる 10 個の乱数のシードを元に生成した. つまり, 合計 40 種類の数値データを用意した.

2.4 モデル

モデルには, OpenAI 社製 3 種類, Google 社製 5 種類, Anthropic 社製 2 種類, Meta 社製 3 種類, Misral AI 社製 2 種類の異なる性能をもつ合計 15 種類の LLM を用いた. LLM の利用は全て API を通して行った. モデルの詳細の一覧と利用した API については, 付録 A.2 に示した.

3 評価

実験で得られた LLM の回答が, 基本的な統計量等の説明となっているか, 1) 統計量に関する単語の出

現率, 2) 説明された統計量の精度, 3) 例示と補足の効果とコストの点で評価した.

3.1 統計量に関する単語の出現率

LLM の回答が基本的な統計量等の説明となっているか, 基本的な統計量の単語を参照単語 (Reference word) として定義し, 参照単語が LLM の回答にどの程度含まれるかを評価した. 具体的には, 統計分析の基本である平均, 標準偏差, 最小値, 中央値, 最大値, 四分位に対応する英単語とその類義語を数え上げ, 数式 (1) に示した Reference word emerge rate を求めた.

$$\frac{\text{回答に含まれた参照単語の数}}{\text{定義した参照単語の数}} \tag{1}$$

なお, 参照単語は見出し語に変換し, 完全一致で数え上げた. 具体的な参照単語と類義語については, 付録 A.3 に示した.

3.2 説明された統計量の精度

LLM の回答で示された統計量の数値が, 一定の誤差範囲内であることを検証するため, データセットから求めた統計量 (Ground truth) と LLM の回答の統計量との差を評価した. 具体的には, 参照単語に対応する統計量の数値を LLM の回答から抽出し, 数式 (2) に示した Ground truth residual rate を求めた.

$$\frac{\text{Ground truth と LLM の回答の差の絶対値}}{\text{Ground truth の値}} \tag{2}$$

なお, LLM の回答から統計量を抽出する処理は, Google 社製の Gemini 1.5 flash を用いた.

3.3 例示と補足の効果とコスト

プロンプト条件ごと, 特に Example 条件と Supplement 条件の説明能力を向上させる効果をモデルごとに比較するため, 次の方法で同等性の可視化と統計検定を行い, コストについても評価した.

同等性の可視化 Example 条件と Supplement 条件の Reference word emerge rate を比較することで, Example 条件あたりの Supplement 条件の説明能力を向上させる効果をモデルごとに比較した.

同等性の検定 Example 条件と Baseline 条件及び Example 条件と Supplement 条件の Reference word emerge rate の平均値の差を, 信頼区間法を用いた同等性検定 [22, 23] によって評価した. 信頼区間は 90%, 同等性マージンは参照単語として用いた 6 種類の単語のうち 1 単語分にあたる ±1/6 とした.

コストの比較 Example 条件と Supplement 条件のコストについて, 各条件のプロンプトの入力文字数, 正確には API 利用コスト計算に用いられる Input token 数をコストと定義して評価した. Input token 数の集計には, OpenAI 社の Tokenizer（入力文字から Token を出力するプログラム）を用いた. Token 数がより少なければ API 利用コストも低くなり, より説明におけるコストが低いと言える.

4 結果

4.1 統計量に関する単語の出現率

図 2 に条件/モデルごとの Reference word emerge rate を示す. 例示や補足のない Baseline 条件よりも, 基本的な統計量の例示を行った Example 条件や LLM の役割や分析の目的等の補足を行った Supplement 条件の方が, Reference word emerge rate が高かった. また, Supplement 条件と Example 条件の Reference word emerge rate は, Baseline 条件と比較して近かった.

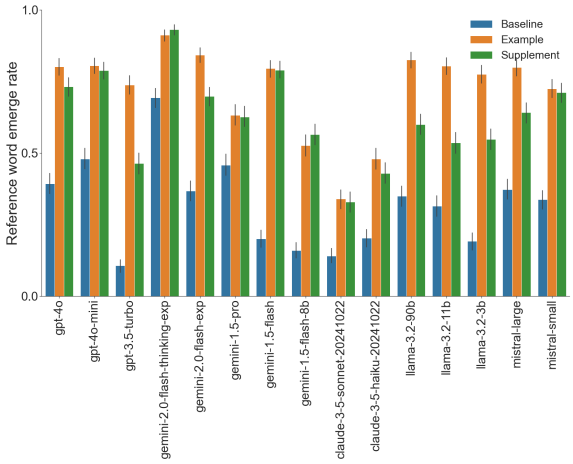

図 2 条件/モデルごとの Reference word emerge rate

4.2 説明された統計量の精度

Ground truth residual rate は, 全てのモデル, 条件を合わせて, 平均 0.10, 標準偏差 1.44 となった.

4.3 例示と補足の効果とコスト

同等性の可視化 図 3 に Supplement/Example 条件間における Reference word emerge rate を示す. x 軸と y 軸が, Example 条件と Supplement 条件の Reference word emerge rate である. 散布図において, 両条件の結果が同等な場合は 45 度線に近くなり, 異なる場合はその線から離れる. Supplement 条件と Example 条件の Reference word emerge rate は一部のモデルを除き,

45 度線に誤差範囲内で重なり, 比例関係にあった.

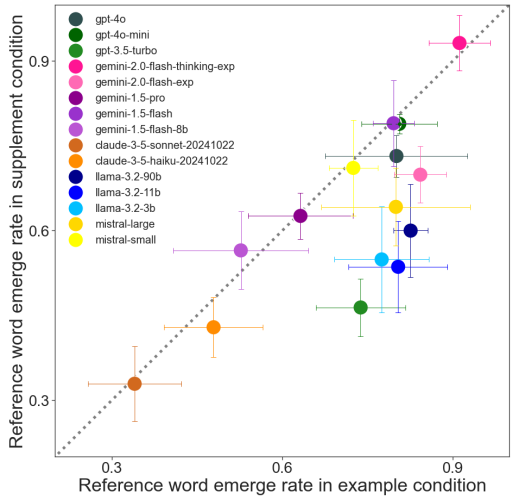

図 3 Supplement/Example 条件間の Reference word emerge rate

同等性の検定 Example 条件と Supplement 条件の Reference word emerge rate の平均値の差に関する同等性検定の結果, 両条件は全 15 モデル中 9 モデルにおいて同等であった. 同等なモデルについては表 2 に示す. 一方で, Example 条件と Baseline 条件で同等なモデルは無かった.

表 2 例示と補足での説明能力が同等と言えるモデル

モデル名	平均の差	信頼区間 下限	上限
gemini-1.5-flash	0.004	-0.04	0.048
gemini-1.5-pro	0.006	-0.046	0.058
claude-3-5-sonnet	0.011	-0.015	0.037
mistral-small	0.014	-0.019	0.047
gpt-4o-mini	0.017	-0.013	0.047
gemini-2.0-thinking	-0.020	-0.039	-0.001
gemini-1.5-flash-8b	-0.045	-0.109	0.019
claude-3-5-haiku	0.05	-0.001	0.101
gpt-4o	0.069	0.033	0.105

コストの比較 Input token 数は, Example 条件で平均 153.08, 標準偏差 87.21, Supplement 条件で平均 35.33, 標準偏差 8.22 となった. いずれの条件でもサンプル数は 120 であった. これらの結果に対して, Welch の t 検定を行ったところ, Supplement 条件の方が, Example 条件より有意に Input token 数が少ないことが示された (t_{238}=14.66, $p < 0.001$).

5 考察

本研究の目的は, LLM による定性的な推論と定量的な推論を組み合わせた数値データの説明において, 統計知識に基づいた説明能力を向上させることが可能な手法を明らかにすることであった. そのため, 3 つの指示条件 (i.e., 例示も補足もない Baseline 条件, 基本的な統計量等を例示した Example 条件, LLM の役割や分析の目的等について補足した Supplement 条件) の説明能力を比較した. Reference word emerge rate で評価した説明能力は, Baseline 条件よりも, Example 条件と Supplement 条件の方が高く, 例示を行う場合と同等の説明能力を, LLM の役割や分析の目的等の補足をすることで実現できることが分かった. さらに, モデルごとに説明能力を比較すると, 例示で説明能力が高くなるモデルは, 補足をすることでも説明能力が高くなることが分かった. また, 数値データの説明能力が高いのは, 必ずしもパラメータ数の多い（一般的に性能の高い）上位モデルではなく, 中位モデルとなった. Ground truth residual rate で評価した定量的な統計量の説明能力については, 10%程度の誤差で説明されており, 数値データの説明を定量的にも行うことができた. さらに, Input token 数で評価したコストについては, 例示を行う場合よりも, LLM の役割や分析の目的等の補足をする場合の方が, コストが低いことが確かめられた. 実験的には Input token 数, すなわち API 利用コストによってコストを評価したが, データの説明についての例示を行うには, LLM 利用者のデータ分析や統計に関する専門知識の学習コストが必要となる. すなわち, 例示を行うよりも LLM の役割や分析の目的等の補足をする方が, LLM 利用者の学習コストという点でもコストが低いと考えられる.

本研究では, 数値データの統計知識に基づいた説明において, LLM の役割や分析の目的等の補足をするという費用対効果の高い方法と, 数値データの説明能力の高いモデルを明らかにした. これらの成果は, 実務でも活用できると考えられる. 一方で, 一変量の 100 データポイントの数値データの基本的な統計量での説明課題を用いたが, データセットや統計量が複雑になった場合, 補足と比べて例示の能力が高くなる可能性も否定できない. 今後はデータセットや統計量の種類を増やし, 研究を発展させたい.

6 結論

本研究により, 基本的な統計量等の例示よりも, LLM の役割や分析の目的等の補足の方が, 説明能力を向上させる効果は同等でも, コストは低くできることが分かった. これにより, LLM の訓練コストのない文脈内学習において, 定性的な推論と定量的な推論の双方が必要な, 数値データの統計知識に基づいた説明能力を向上させる手法を明らかにした.

参考文献

[1] Dan Hendrycks, Collin Burns, Steven Basart, Andy Zou, Mantas Mazeika, Dawn Song, and Jacob Steinhardt. Measuring massive multitask language understanding. In **International Conference on Learning Representations**, 2021.

[2] Dan Hendrycks, Collin Burns, Saurav Kadavath, Akul Arora, Steven Basart, Eric Tang, Dawn Song, and Jacob Steinhardt. Measuring mathematical problem solving with the MATH dataset. In **Thirty-fifth Conference on Neural Information Processing Systems Datasets and Benchmarks Track (Round 2)**, 2021.

[3] Karl Cobbe, Vineet Kosaraju, Mohammad Bavarian, Mark Chen, Heewoo Jun, Lukasz Kaiser, Matthias Plappert, Jerry Tworek, Jacob Hilton, Reiichiro Nakano, et al. Training verifiers to solve math word problems. **arXiv preprint arXiv:2110.14168**, 2021.

[4] Hao Xue and Flora D Salim. Promptcast: A new prompt-based learning paradigm for time series forecasting. **IEEE Transactions on Knowledge and Data Engineering**, 2023.

[5] Nate Gruver, Marc Finzi, Shikai Qiu, and Andrew G Wilson. Large language models are zero-shot time series forecasters. **Advances in Neural Information Processing Systems**, Vol. 36, , 2024.

[6] Shankar Kantharaj, Rixie Tiffany Leong, Xiang Lin, Ahmed Masry, Megh Thakkar, Enamul Hoque, and Shafiq Joty. Chart-to-text: A large-scale benchmark for chart summarization. In **Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)**, pp. 4005–4023, Dublin, Ireland, May 2022. Association for Computational Linguistics.

[7] Y Hu et al. SportsMetrics: Blending text and numerical data to understand information fusion in LLMs. In **Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)**, pp. 267–278, Bangkok, Thailand, August 2024. Association for Computational Linguistics.

[8] S Murakami et al. Learning to generate market comments from stock prices. In **Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)**, pp. 1374–1384, Vancouver, Canada, July 2017. Association for Computational Linguistics.

[9] Yizhang Zhu, Shiyin Du, Boyan Li, Yuyu Luo, and Nan Tang. Are large language models good statisticians? In **The Thirty-eight Conference on Neural Information Processing Systems Datasets and Benchmarks Track**, 2024.

[10] Chan et al. Mle-bench: Evaluating machine learning agents on machine learning engineering. **arXiv preprint arXiv:2410.07095**, 2024.

[11] M Bodhisattwa et al. Data-driven discovery with large generative models. **arXiv preprint arXiv:2402.13610**, 2024.

[12] Chris Lu, Cong Lu, Robert Tjarko Lange, Jakob Foerster, Jeff Clune, and David Ha. The ai scientist: Towards fully automated open-ended scientific discovery. **arXiv preprint arXiv:2408.06292**, 2024.

[13] Ziru Chen et al. Scienceagentbench: Toward rigorous assessment of language agents for data-driven scientific discovery. **arXiv preprint arXiv:2410.05080**, 2024.

[14] T Brown et al. Language models are few-shot learners. In **Advances in Neural Information Processing Systems**, Vol. 33, pp. 1877–1901. Curran Associates, Inc., 2020.

[15] J Wei et al. Emergent abilities of large language models. **Transactions on Machine Learning Research**, 2022.

[16] Z Wang et al. RoleLLM: Benchmarking, eliciting, and enhancing role-playing abilities of large language models. In **Findings of the Association for Computational Linguistics: ACL 2024**, pp. 14743–14777, Bangkok, Thailand, August 2024. Association for Computational Linguistics.

[17] Aobo Kong, Shiwan Zhao, Hao Chen, Qicheng Li, Yong Qin, Ruiqi Sun, Xin Zhou, Enzhi Wang, and Xiaohang Dong. Better zero-shot reasoning with role-play prompting. In **Proceedings of the 2024 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (Volume 1: Long Papers)**, pp. 4099–4113, Mexico City, Mexico, June 2024. Association for Computational Linguistics.

[18] Mingqian Zheng, Jiaxin Pei, Lajanugen Logeswaran, Moontae Lee, and David" Jurgens. When "a helpful assistant" is not really helpful: Personas in system prompts do not improve performances of large language models. In **Findings of the Association for Computational Linguistics: EMNLP 2024**, pp. 15126–15154, Miami, Florida, USA, November 2024. Association for Computational Linguistics.

[19] Xinlei Wang, Maike Feng, Jing Qiu, Jinjin Gu, and Junhua Zhao. From news to forecast: Integrating event analysis in LLM-based time series forecasting with reflection. In **The Thirty-eighth Annual Conference on Neural Information Processing Systems**, 2024.

[20] Andrew Robert Williams, Arjun Ashok, Étienne Marcotte, Valentina Zantedeschi, Jithendaraa Subramanian, Roland Riachi, James Requeima, Alexandre Lacoste, Irina Rish, Nicolas Chapados, and Alexandre Drouin. Context is key: A benchmark for forecasting with essential textual information. **arXiv preprint arXiv:2410.18959**, 2024.

[21] Charles R. C Harris Harris, K. Jarrod Millman, Stéfan J. van der Walt, Ralf Gommers, Pauli Virtanen, David Cournapeau, and Wieser et al. Array programming with NumPy. **Nature**, Vol. 585, No. 7825, pp. 357–362, September 2020.

[22] E Garbe, J Röhmel, and U Gundert-Remy. Clinical and statistical issues in therapeutic equivalence trials. **European journal of clinical pharmacology**, Vol. 45, pp. 1–7, 1993.

[23] Daniël Lakens. Equivalence tests: A practical primer for t tests, correlations, and meta-analyses. **Social psychological and personality science**, Vol. 8, No. 4, pp. 355–362, 2017.

A 付録

A.1 プロンプトの指示部の一覧

条件名	プロンプト名	プロンプト例
Baseline	Comprehensive	`Please describe comprehensively what you can read from the following data in CSV format with specific values.`
	Brief	`Please describe important points briefly ( about 50 words ) what you can read from the following data in CSV format with specific values.`
	Important	`Please describe important points briefly what you can read from the following data in CSV format with specific values.`

Example	One shot	`Please describe comprehensively what you can read from the following data in CSV format with specific values.`
		`Especially describe and calculate the following points.`
		`## Example`
		`### Input`
		`10.254,12.804,10.63,8.283,9.468,8.702,13.133,5.817,12.913,11.891,...`
		`### Outout`
		`- count of the dataset: 20`
		`- mean of the dataset: 10.238`
		`- standard deviation of the dataset: 1.983`

	Instructional	`Please describe comprehensively what you can read from the following data in CSV format with specific values.`
		`Especially describe and calculate the following points.`
		`- count of the dataset`
		`- mean of the dataset`

	Chain of thought	`Please describe comprehensively what you can read from the following data in CSV format with specific values.`
		`- First, understand data representation formats.`
		`- Second, analyze data patterns based on the representation formats.`
		`- Third, identify or calculate key values or statistics from the data.`
		`- Finally, summarize and describe the important aspects of the data.`

Supplement	Description	`This is comma separated values of univariate data.` `Please describe comprehensively what you can read from the following data with specific values.`
	Persona	`You are a professional data analyst, responsible for interpreting data with expertise and reporting it clearly and concisely in a way that anyone can easily understand.` `Please describe comprehensively what you can read from the following data with specific values.`
	Goal	`The goal of analysis is to understand patterns or identify important numbers and statistics in the dataset.` `Please describe comprehensively what you can read from the following data with specific values.`

A.2 LLM モデルの一覧

モデルには以下の API から接続を行った.

開発元	モデル名	API
OpenAI	gpt-4o, gpt-4o-mini, gpt-3.5-turbo	OpenAI API (Batch)
Google	gemini-2.0-flash-thinking-exp, gemini-2.0-flash-exp gemini-1.5-pro, gemini-1.5-flash, gemini-1.5-flash-8b	Google Gemini API
Anthropic	claude-3-5-sonnet-20241022, claude-3-5-haiku-20241022	Anthropic API (Batch)
Meta	llama-3.2-90b, llama-3.2-11b, llama-3.2-3b	Amazon Bedrock API
Mistral AI	mistral-large, mistral-small	Amazon Bedrock API

A.3 統計量の参照単語と Ground truth の計算方法

参照単語は類義語も同単語として判定した. Ground truth の計算は Python の NumPy を利用した.

参照単語	類義語	Ground truth の計算方法
mean	average	`np.mean()` で平均値を計算
deviation	-	`np.std()` で標準偏差を計算
median	-	`np.percentile()` で **50%**パーセンタイルを計算
percentile	quantile	`np.percentile()` で **25%**と **75%**パーセンタイルを計算
minimum	min	`np.percentile()` で **0%**パーセンタイルを計算
maximum	max	`np.percentile()` で **100%**パーセンタイルを計算