

# 文書質問応答データセットの質問応答特性を保持した自動拡張

小島亮<sup>1</sup> 三輪誠<sup>1,2</sup>

<sup>1</sup> 豊田工業大学 <sup>2</sup> 産業技術総合研究所人工知能研究センター  
{sd22041,makoto-miwa}@toyota-ti.ac.jp

## 概要

文書質問応答におけるデータ汚染の問題に対処するため、既存の評価用データセットにおける質問タイプ、ドメイン、時間的制約などの質問応答の特性 (QA 特性) を保持しながら、大規模言語モデル (LLM) を用いて自動拡張する手法を提案する。提案手法は3段階で構成される。まず、評価用データセットの設計において考慮されていると考えられる QA 特性を抽出する。次に、QA 特性の組み合わせをスキーマとして明示的に構造化する。最後に、新たな文書についてスキーマに基づいた質問応答データを生成する。既存の CRAG データセットを用いた実験により、提案手法は Few-shot 手法による拡張と比較して元データセットの QA 特性をより良く保持できることを確認した。

## 1 はじめに

言語処理システムの性能評価のため、与えられた文書に基づいて質問に回答する文書質問応答 (Document Question Answering; DocQA) が広く用いられている [1, 2, 3]。DocQA における評価データセットでは、評価の観点 (知識理解, 推論, 情報統合など) に応じて、質問タイプや制約など様々な特性を考慮して、それぞれの質問応答が作成されている。この特性には、質問が扱う分野 (映画, スポーツなど)、質問の種類 (事実抽出, 比較, 集計など)、回答に必要な情報の時間的性質 (静的事実, 変化する情報など) などが挙げられる。本研究では、これらのカテゴリとその取りうる値を質問応答特性 (QA 特性) と定義する。

大規模言語モデル (LLM) は継続的に更新されるため、その性能評価も継続的に行うことが望ましい。しかし、この継続的な評価において、データ汚染の問題が存在する [4, 5]。データ汚染とは、評価用データセットが LLM の学習データに含まれてしまい、その学習に基づいて回答する現象である。

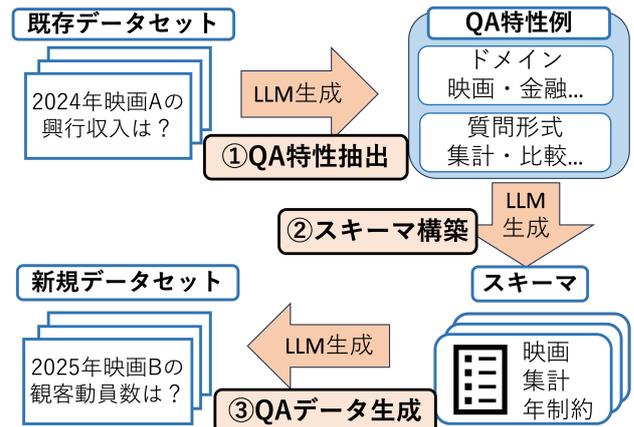


図1 提案手法の概要

データ汚染が発生すると信頼性のある評価が困難になるため、LLM の学習データに含まれない評価用データセットを新たに作成し直す必要がある。

この課題を解決するために、既存のデータセットをもとに新たなデータセットを作成するデータセット拡張手法が提案されている。このような手法には、既存のデータを少数例示して新たなデータを生成する Few-shot 手法 [6, 7] がある。しかし、Few-shot 手法では LLM が例示から暗黙的に特性を獲得するため、どの特性が保持され、どの特性が失われるかを予測・制御できない。その結果、生成されるデータセットでは元データセットの QA 特性が十分に保持されず、既存データセットが意図した能力の評価が困難になる可能性がある。

本研究では、このような問題に対処するため、DocQA における QA 特性を保持したデータセット自動拡張を目指す。図1に提案手法の概要を示す。提案手法は、LLM を基盤に、(1) QA 特性抽出、(2) スキーマ構築、(3) QA データ生成の3段階で自動拡張を行う。QA 特性抽出では、既存 QA データセット全体を分析し、データセットに含まれる QA 特性を抽出する。スキーマ構築では、元データにある QA 特性の値の組み合わせをスキーマとして明示的に構造化する。QA データ生成では、新たな文書に基

づいてスキーマに適合した QA を生成する。CRAG データセット [8] を用いた実験により、提案手法は Few-shot 手法と比較して元データセットの QA 特性をより良く保持できることを確認した。

本研究における貢献は以下の 3 点である。

- 既存 DocQA データセットの QA 特性を QA データから抽出し、それらの QA 特性を保持した新規データセットを作成する LLM を用いたデータセット自動拡張手法を提案した。
- 元の評価データセットに含まれない新規文書に基づいた QA データ生成を可能にし、データ汚染問題に対処しながら QA 特性を保持する評価データセット拡張を実現した。
- CRAG を対象とした実験において、提案手法が Few-shot 手法と比較して元データセットの QA 特性をより良く保持できることを示した。

## 2 関連研究

### 2.1 DocQA 評価データセット

DocQA の評価データセットは、与えられた文書の情報に基づいて回答を生成するシステムの能力を評価するために用いられ、様々なデータセットが提案されている [1, 2, 3]。これらのデータセットは、質問に関連する文書を外部知識から検索し、その文書に基づいて LLM が回答を生成する検索拡張生成 (Retrieval Augmented Generation; RAG) の評価において特に重要である。RAG の包括的な評価を目指した CRAG (Comprehensive RAG Benchmark) [8] では、各 QA に 8 種類の質問タイプ、5 種類のドメイン、4 種類の時間的制約からなるメタデータが付与されており、システムの様々な側面を詳細に評価できる。

### 2.2 LLM を用いたデータセット拡張

LLM を用いたデータセット拡張手法として、Few-shot 手法が提案されている [6, 7]。Few-shot 手法では、既存のデータを例示としてプロンプトに含め、新たなデータを生成する。この手法は人手でのデータセット作成と比較して実装が容易かつ低コストである一方、LLM が例示から暗黙的に特性を獲得するため、どの特性が保持され、どの特性が失われるかを予測・制御できない。この課題に対し、RAGEval [7] は文書生成においてスキーマを利用している。RAGEval では、専門分野の文書に含まれる

表 1 スキーマの構造

要素	内容
スキーマ ID	各スキーマの識別子
QA 特性の組み合わせ説明	各カテゴリに対する具体的な値 例：ドメイン：映画，質問意図：比較
意図する QA	意図する QA データの概要
実例 QA	該当する QA データの例示

べき情報や構造をスキーマとして定義し、文書生成の品質と一貫性を向上させている。しかし、QA 生成では 7 種類の質問形式を定義した Few-shot 手法に留まっており、より詳細な QA データの特性を捉えた制御は行われていない。

## 3 提案手法

本研究では、既存の DocQA データセットにおける QA 特性を抽出し、その QA 特性を保持した新規データセットを生成することで、データセットを自動拡張する手法を提案する。本研究における QA 特性とは、評価データセットの対象とする評価の観点 (知識理解、推論、情報統合など) を反映するために、作成時に暗黙的に考慮されたと考えられる QA データの特性であり、ドメイン、質問の種類、時間的制約などの **カテゴリ** とドメインにおける「映画」「スポーツ」など **カテゴリ** に対応する具体的な **値** の組を指す。提案手法は、LLM を基盤に、QA 特性抽出、スキーマ構築、QA データ生成の 3 段階で自動拡張を行う。

### 3.1 QA 特性抽出

QA 特性抽出では、LLM を用いて既存 QA データセット全体を分析し、データセットに含まれる QA 特性のカテゴリとその取りうる値の候補を抽出する。多くのベンチマークでは QA 特性が付与されていないため、QA データ生成に先立ち、評価用データセットの設計において考慮されたと考えられる QA 特性を特定し、抽出する必要がある。この段階では、個々の QA データではなくデータセット全体を分析し、複数の QA データ間に共通して観察される特徴を QA 特性として抽出する。

### 3.2 スキーマ構築

スキーマ構築では、抽出された QA 特性に基づいて、LLM を用いて元データセットに存在する QA 特性の組み合わせとして、QA データを生成する際の設計図であるスキーマを構築する。スキーマの構造を表 1 に示す。各スキーマは、スキーマ ID、QA 特

性の組み合わせ、説明、実例 QA から構成される。本研究では、スキーマを構築する元データの単位が異なる2つのスキーマ構築アプローチを比較する。

### 3.2.1 個別 QA データからのスキーマ構築

各 QA データを個別に分析し、その QA データに現れる QA 特性を1件のスキーマとして生成する。QA データ数と同数のスキーマを構築し、個別の QA データの特性をより忠実に反映する。

### 3.2.2 データセット全体からのスキーマ構築

既存 QA データセット全体を分析し、類似した QA 特性を持つ QA データをまとめて少数のスキーマを構築する。プロンプトでは、データセット全体の QA 特性の組み合わせパターンを捉えるために必要な数のスキーマを生成するよう指示する。スキーマ構築の後、LLM に元データセット全体を入力してスキーマごとに分類し、その割合に基づいて生成数を決定する。

## 3.3 QA データ生成

QA データ生成では、スキーマと文書を用いて新規 QA データを生成する。生成時には、QA データ生成の前に、文書-スキーマの適合性判定を行い、スキーマの要件を満たす QA データのみを生成するようにする。この適合性判定では、文書とスキーマを LLM に入力し、スキーマの要件を満たす QA データが生成可能か判定する。この適合性判定で生成可能と判定されたスキーマと文書について、QA データを生成する。その後、生成された QA データが参照文書から回答可能かを評価するため、QA データと参照文書を LLM に入力し、参照文書の情報のみで質問に回答できるかを判定し、回答不可能と判定された QA データは再生成を実施する。

## 4 実験

### 4.1 実験設定

DocQA 評価データセットは次の手順で作成した。まず、RAG 用の評価データセットである CRAG [8] から 1,000 件の QA データをサンプリングした。この際、本研究は参照文書に基づいて回答可能な QA の生成を目的としているため、質問タイプが回答の存在しない False Premise である質問は対象外とした。次に、CRAG から検索の要素を除外し、参照

表 2 質問タイプ分類結果 (%)

質問タイプ	元	全体	個別	FS5	FS1
Simple	33	31	35	55	46
Simple w. Cond.	27	33	26	25	29
Set	7	13	10	3	4
Comparison	5	5	3	0	2
Aggregation	19	16	16	15	13
Multi-hop	3	1	0	2	2
Post-proc.	6	1	10	0	4

表 3 RAG 性能評価 (正解率%)

手法	参照なし	Gold	BM25
元データ	69	79	77
Few-shot(5)	67	99	96
Few-shot(1)	65	99	96
提案 (全体)	60	98	92
提案 (個別)	58	97	93

文書と対応のある DocQA データセットを作成した。具体的には、1,000 件の QA それぞれに対して、CRAG データセットに保存されている検索結果上位 10 件の HTML 文書を取得し、本文抽出ライブラリ Trafilatura [9] を用いてテキストを抽出し、10 件の文書を作成した。その後、各文書と QA の組み合わせについて、LLM を用いて、文書のみで回答可能かを判定し、回答可能と判定された 557 件を抽出し、DocQA データセットを構築した。

実験では、データセットのうち 200 件のみを利用し、そのうちの 100 件を提案手法における QA 特性抽出・スキーマ構築および Few-shot 手法における例示データとして使用し、残りの 100 件を QA 生成用の文書コーパスとして使用した。

データセット拡張には、以下の 4 手法を用いた。すべての手法において、ランダムに選択された 1 件の文書を入力とし、QA データを 1 件生成した。

- **Few-shot 5 件 (FS5)** : ランダムに抽出した 5 件の QA データをプロンプトに例示。
- **Few-shot 1 件 (FS1)** : ランダムに抽出した 1 件の QA データをプロンプトに例示。
- **提案手法 (全体)** : データセット全体から構築したスキーマ 1 件をプロンプトに例示。
- **提案手法 (個別)** : 個別 QA から構築したスキーマ 1 件をプロンプトに例示。

各手法により 100 件の QA データセットを生成した。データセット拡張およびその LLM 評価には、すべて Gemini-2.5-flash [10] を LLM として使用した。データセットの詳細については付録 A に記載する。

表 4 Black Swan に関する文書における生成事例の比較

手法	参照種別	参照内容	生成結果	分類
FS5	例示 QA	Q: is the eastern honey bee or the western honey bee more common? A: the western honey bee... ( <i>Comparison</i> <sup>†</sup> ) Q: how many albums has lady gaga released? A: seven studio albums... ( <i>Aggregation</i> <sup>†</sup> )	Q: Who directed the film Black Swan? A: Darren Aronofsky directed Black Swan.	Simple
提案 (全体)	スキーマ	質問意図: Comparative 推論タイプ: Quantitative_Comparison 回答形式: Atomic ドメイン: Arts_Entertainment_Media	Q: Which film critic association awarded more accolades to Black Swan, Austin or Independent Spirit? A: Austin Film Critics Association	Comparison

† 参考として、例示 QA には質問タイプ分類を示した。

## 4.2 QA 特性保持の評価

生成された QA データの QA 特性保持を検証するため、QA 特性の一例である質問タイプ分布を比較した。質問タイプの定義には、CRAG で定義されている 7 種類の質問タイプを使用し、LLM により分類した (詳細な定義は付録 B.1)。スキーマ構築に使用した 100 件の QA データ (元データ) と生成した各 100 件の QA データを対象とする。分類結果を表 2 に示す。LLM による分類の妥当性は人手で検証し、Fleiss' Kappa = 0.70 となった (詳細は付録 B.2)。

Few-shot 手法では、単純な質問パターンである Simple の割合が 46%~55% と高く、元データ (33%) から大きく乖離した。また、複雑な質問パターンである Set や Comparison は元データより低かった。この Few-shot 手法における偏りは、LLM が例示から暗黙的に特性を獲得する際に、単純な質問に偏りやすいことを示唆している。

一方、提案手法は全体および個別 QA の双方で元データの分布により近い結果を示した。ただし、複数の推論ステップを要する Multi-hop の割合は両手法とも低く、これは単一文書からの生成という制約に起因すると考えられる。

## 4.3 RAG 性能評価

生成されたデータセットの実用性を検証するため、RAG による回答生成の性能評価を行った。文書コーパスとして使用した 100 件の QA データ (元データ) と各手法で生成した各 100 件の QA データを対象とする。評価条件として、参照なし (LLM の事前知識のみ)、Gold (正解文書を直接付与)、BM25 (BM25 [11] による検索で上位 5 件の文書を取得し回答) の 3 条件を設定した。正解判定には LLM-as-a-Judge を採用した (無作為抽出した 50 件における LLM と学生 2 名の Fleiss' Kappa は 0.89)。

結果を表 3 に示す。

Gold 条件では、提案手法および Few-shot 手法ともに元データ (79%) より大幅に高い正解率を示した。この差異の要因を分析するため、不正解と判定された 21 件を人手で分類した。その結果、13 件は質問と文書間の時間基準や定義の差異による解釈の不一致に起因していた。加えて、6 件は時間経過により変化する情報の抽出失敗が主な要因であった。生成手法は文書から直接 QA データを作成するため、齟齬がなく、高い正解率を示したと考えられる。

BM25 条件では、提案手法 (92%~93%) が Few-shot 手法 (96%) より低く、検索の難しい QA を生成していることが示唆されるが、元データ (77%) との乖離は依然として見られる。

## 4.4 事例分析

表 4 に同一文書から各手法で生成された事例を示す。Few-shot 手法の例示には比較質問や集計質問が含まれていたが、生成結果は単純な質問パターン Simple となった。一方、提案手法ではスキーマで Comparative を明示的に指定したため、比較質問が生成された。この結果は、Few-shot 手法では例示に含まれる QA 特性が暗黙的であり、生成時に保持されず、提案手法における明示的な QA 特性の制御が必要であることを示唆している。

## 5 おわりに

DocQA 評価データセットの QA 特性を保持したデータセット自動拡張手法を提案した。CRAG データセットを用いた実験により、提案手法は Few-shot 手法と比較して元データセットの QA 特性をより忠実に再現できることを確認した。一方で、RAG 性能評価において、生成されたデータセットは元データセットと比較して高い正解率を示し、難易度に乖離が見られ、この制御は今後の課題である。

## 謝辞

この成果の一部は、NEDO（国立研究開発法人新エネルギー・産業技術総合開発機構）の委託業務（JPNP25006）の結果得られたものです。本研究に関して有益なご議論とご助言をいただきました産業技術総合研究所の浅田真生氏に深く感謝いたします。

## 参考文献

- [1] Pranav Rajpurkar, Jian Zhang, Konstantin Lopyrev, and Percy Liang. SQuAD: 100,000+ questions for machine comprehension of text. In **Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing**, pp. 2383–2392, Austin, Texas, November 2016. Association for Computational Linguistics.
- [2] Tom Kwiatkowski, Jennimaria Palomaki, Olivia Redfield, Michael Collins, Ankur Parikh, Chris Alberti, Danielle Epstein, Illia Polosukhin, Jacob Devlin, Kenton Lee, Kristina Toutanova, Llion Jones, Matthew Kelcey, Ming-Wei Chang, Andrew M. Dai, Jakob Uszkoreit, Quoc Le, and Slav Petrov. Natural questions: A benchmark for question answering research. **Transactions of the Association for Computational Linguistics**, Vol. 7, pp. 452–466, 2019.
- [3] Zhilin Yang, Peng Qi, Saizheng Zhang, Yoshua Bengio, William Cohen, Ruslan Salakhutdinov, and Christopher D. Manning. HotpotQA: A dataset for diverse, explainable multi-hop question answering. In Ellen Riloff, David Chiang, Julia Hockenmaier, and Jun’ichi Tsujii, editors, **Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing**, pp. 2369–2380, Brussels, Belgium, October–November 2018. Association for Computational Linguistics.
- [4] Yajuan Fu, Ozlem Uzuner, Meliha Yetisgen, and Fei Xia. Does data contamination detection work (well) for LLMs? a survey and evaluation on detection assumptions. In Luis Chiruzzo, Alan Ritter, and Lu Wang, editors, **Findings of the Association for Computational Linguistics: NAACL 2025**, pp. 5235–5256, Albuquerque, New Mexico, April 2025. Association for Computational Linguistics.
- [5] Shahriar Golchin and Mihai Surdeanu. Time travel in LLMs: Tracing data contamination in large language models. In **The Twelfth International Conference on Learning Representations**, 2024.
- [6] Jiahao Ying, Yixin Cao, Yushi Bai, Qianru Sun, Bo Wang, Wei Tang, Zhaojun Ding, Yizhe Yang, Xuanjing Huang, and Shuicheng YAN. Automating dataset updates towards reliable and timely evaluation of large language models. In **The Thirty-eight Conference on Neural Information Processing Systems Datasets and Benchmarks Track**, 2024.
- [7] Kunlun Zhu, Yifan Luo, Dingling Xu, Yukun Yan, Zhenghao Liu, Shi Yu, Ruobing Wang, Shuo Wang, Yishan Li, Nan Zhang, Xu Han, Zhiyuan Liu, and Maosong Sun. RAGEval: Scenario Specific RAG Evaluation Dataset Generation Framework. In **ACL**, 2025.
- [8] Xiao Yang, Kai Sun, Hao Xin, Yushi Sun, Nikita Bhalla, Xiangsen Chen, Sajal Choudhary, Rongze Daniel Gui, Ziran Will Jiang, Ziyu Jiang, Lingkun Kong, Brian Moran, Jiaqi Wang, Yifan Ethan Xu, An Yan, Chenyu Yang, Eting Yuan, Hanwen Zha, Nan Tang, Lei Chen, Nicolas Scheffer, Yue Liu, Nirav Shah, Rakesh Wanga, Anuj Kumar, Wen tau Yih, and Xin Luna Dong. CRAG – Comprehensive RAG Benchmark. In **NeurIPS**, 2024.
- [9] Adrien Barbaresi. Trafalatura: A web scraping library and command-line tool for text discovery and extraction. In Heng Ji, Jong C. Park, and Rui Xia, editors, **Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing: System Demonstrations**, pp. 122–131, Online, August 2021. Association for Computational Linguistics.
- [10] Google Gemini Team. Gemini: A family of highly capable multimodal models. **arXiv preprint arXiv:2312.11805**, 2023.
- [11] Stephen Robertson and Hugo Zaragoza. The probabilistic relevance framework: Bm25 and beyond, April 2009.

## A 実験詳細

### A.1 使用モデルとパラメータ

全ての実験において、LLM として Gemini-2.5-flash を使用した。パラメータは temperature=1, top\_p=1, seed=0, max\_output\_tokens=65,535 とした。

### A.2 各手法の設定

各手法の設定を表 5 に示す。

	FS5	FS1	全体	個別
参照種別	QA	QA	スキーマ	スキーマ
参照数	5 件	1 件	1 件	1 件
スキーマ総数	—	—	22	100
文書適合判定	—	—	有	有
生成 QA 数	100	100	100	100

### A.3 提案手法の詳細

#### A.3.1 抽出された QA 特性

QA 特性抽出結果を表 6 に示す。

表 6 抽出された QA 特性のカテゴリと値数

カテゴリ	全体	個別
質問意図	10 種	7 種
回答形式	6 種	6 種
推論タイプ	9 種	5 種
ドメイン	8 種	7 種
時間的制約	4 種	3 種
回答長	—	4 種

#### A.3.2 スキーマ分布

提案手法（全体）で生成された 22 件のスキーマに基づいた分類結果（上位 5 件）を表 7 に示す。

表 7 スキーマ分布（上位 5 件）

ID	概要	割合
05	エンティティ識別・単一値	10%
18	特定時点検索・日時形式	9%
01	事実検索・単一値	8%
06	エンティティ識別・文脈文	8%
07	リスト生成・リスト形式	8%

### A.4 スキーマ構築アプローチの比較事例

提案手法（全体）と提案手法（個別）の違いを示す事例を表 8 に示す。データセット全体からのアプローチでは複数の QA から「数量を問う」共通パターンを抽出し集計質問を生成した。一方、個別 QA からのアプローチでは元 QA の「リスト形式で回答」という QA 特性を忠実に再現し Set 質問を生成した。

### A.5 回答可能性評価

QA データと参照文書を LLM に入力し、参照文書の情報のみで質問に回答できるかを判定した。その結果、Few-shot 手法は各データセットで 100%、提案手法は各データセットで 97%が回答可能と判定された。提案手法

表 8 Premier League 文書における生成事例の比較

手法	スキーマ内容	生成結果	分類
提案 (全体)	質問意図: Count_Quantity 推論タイプ: Direct_Aggregation	Q: How many Premier League titles has Manchester United won? A: 13 titles	Agg.
提案 (個別)	質問意図: List_Generation 回答形式: List	Q: Which clubs have been ever-present in the Premier League? A: Arsenal, Chelsea,...	Set

で回答不可能と判定された 3%（6 件）は、再生成を実施した。結果、全ての QA データが回答可能と判断された。

## B 質問タイプ分類詳細

### B.1 質問タイプの定義

本研究で使用した質問タイプの定義を表 9 に示す。

表 9 質問タイプの定義（CRAG [8] に基づく）

質問タイプ	定義
Simple	単純な事実を問う質問
Simple w. Condition	時間やカテゴリの条件付きで単純な事実を問う質問
Set	複数の物体や人物の集合を回答として期待する質問
Comparison	2 つの物事を比較する質問
Aggregation	検索結果の集計を要する質問
Multi-hop	複数情報を連鎖させる質問
Post-processing heavy	検索した情報の推論や処理を必要とする質問

### B.2 質問タイプ分類の妥当性検証

LLM による質問タイプ分類の妥当性を検証するため、50 件（元データ 10 件、生成データ各 10 件）をサンプリングし、自然言語処理を専門とする学生 2 名が独立に分類した。3 者間（LLM と学生 2 名）での Fleiss' Kappa は 0.70、3 者全員一致率は 68%であった。

## C 使用プロンプト

紙面の都合上、主要プロンプトの概要のみ示す（図 2）。

図 2 提案手法で使用する主要プロンプトの概要

<b>[QA 特性抽出プロンプト]</b> <i>Role:</i> You are a specialist in NLP and dataset construction. <i>Task:</i> Analyze the provided QA data from an existing benchmark, define its evaluation characteristics, and create a prompt for schema generation. <i>Instructions:</i> (1) Analyze the input QA data and define all "Evaluation Characteristics" and their "Possible Values." (2) Output a prompt that instructs an LLM to extract specific combination patterns (schemas) from the data. <i>Definition:</i> Evaluation Characteristics are characteristics intentionally designed in the QA data to reflect the capabilities the benchmark intends to measure.
<b>[スキーマ構築プロンプト]</b> <i>Input:</i> QA dataset with "query" and "answer" fields. <i>Context:</i> Schemas will be used for (1) Filtering: determine if a document allows QA creation, (2) Generation: guide new QA pair generation. <i>Schema Definition:</i> A specific combination of values observed in the original data, selected from: Question Intent, Answer Complexity/Form, Reasoning Type, Domain Category, Temporality. <i>Task:</i> Extract combination patterns. Create schemas to capture benchmark diversity. Include up to 5 matching examples per schema.
<b>[QA データ生成プロンプト]</b> <i>Context:</i> It has been predetermined that QA pairs can be created from this page content based on the specified schema. <i>Input:</i> Schema information (JSON), Feasibility judgment, Page content. <i>Task:</i> Create benchmark QA pairs that meet the schema requirements. <i>Output:</i> JSON format: {"query": "...", "answer": "..."}.