日本語 Natural Questions と BoolQ の構築

植松拓也 [1]　王昊 [1]　河原大輔 [1]　柴田知秀 [2]
[1] 早稲田大学　[2] LINE ヤフー株式会社
{takuya1009@akane.,conan1024hao@akane.,dkw@}waseda.jp tomshiba@lycorp.co.jp

概要

頑健な質問応答 (QA) モデルの訓練、評価を行うためには、様々な QA データセットを用意する必要がある。しかし、多様な QA データセットが存在する言語は英語だけであり、日本語においては少数の基本的な QA データセットしか存在しない。本研究では、人間の情報欲求から自然発生する質問からなる Natural Questions (NQ) と BoolQ の日本語版 (JNQ, JBoolQ) を構築する。構築は、検索エンジンのクエリログから日本語の自然な質問文を収集し、クラウドソーシングを利用したアノテーションによって行う。また、JNQ から 2 つのタスク、JBoolQ から 1 つのタスクを定義し、QA モデルを評価する。

1　はじめに

高性能かつ頑健な自然言語処理 (NLP) モデルを構築するためには、様々な質問応答（QA）データセットを用意し、訓練、評価、分析をすることが重要である。QA データセットには抽出型、生成型、多肢選択式など様々な種類があり、それらを解くためには Multi-hop 推論 [1] や実世界知識 [2] など多くの技術・知識が必要となる。Unified QA [3] や FLAN [4] のように様々な QA タスクを統合的に解く研究もあるが、このような統合的な解析が可能なのは英語だけであり、他言語では多様な QA データセットが存在しないため不可能である。日本語には、JSQuAD [5] や JaQuAD [6]、JAQKET [7] などの基本的な QA データセットしか存在しない。

日本語に存在しない重要な QA データセットとして、人間の情報欲求から自然に発生する質問からなる Natural Questions (NQ) [8] がある。例えば SQuAD [9] では、質問をアノテータに作成してもらうため自然な質問ではなく、annotation artifacts [10] が存在するという問題がある。これに対して、NQ では、検索エンジンにユーザが入力したクエリが用いられており、自然な質問と考えられる。

本研究では、翻訳を使用せず、検索エンジンの日本語クエリログを利用して、日本語 Natural Questions (JNQ) を構築する。クエリログからデータセットを構築するために、NQ では訓練されたアノテータが雇用されていたが、JNQ ではコストを低減するためにクラウドソーシングで行う。

JNQ に加えて、NQ から派生した yes/no 質問からなる BoolQ データセット [11] の日本語版である JBoolQ も構築する。JNQ と JBoolQ の例を図 1 に示す。また、JNQ から Long Answer 抽出タスク、Short Answer タスク、JBoolQ から Yes/No Answer 識別タスクを定義し、ベースラインとなるモデルの評価を行った。

2　日本語 Natural Questions

NQ [8] は、文書を読んで自然な質問に答える能力に焦点を当てたデータセットで、質問文、文書、long answer、short answer から構成される。質問文は、検索エンジンのクエリログから収集されている。文書は、Wikipedia の記事を採用し、1 つの質問文に対し 1 つの文書が与えられる。long answer は、答えを推測するのに十分な情報を含んでいる文書中の段落や表などである。short answer は、質問文に対するできるだけ短い答えであり、文書中のスパンである。

日本語 Natural Questions (JNQ) も、NQ と同様に、質問文、文書、long answer、short answer から構成する。質問文は検索エンジンのクエリログから抽出し、文書は日本語 Wikipedia の記事である。long answer、short answer はクラウドソーシング[1)]を利用して取得する。クラウドソーシングを利用することにより、専門家のアノテータを必要とせずに低コスト、かつ、一定の質を担保したデータセットを構築することができる。long answer は、クラウドソーシングで扱うためにタスクを簡単化し、段落のみを対象とする。NQ では文書中に long answer は一つであるという強い制約があるが、実際は文書中に答えを

1）Yahoo!クラウドソーシングを用いた。

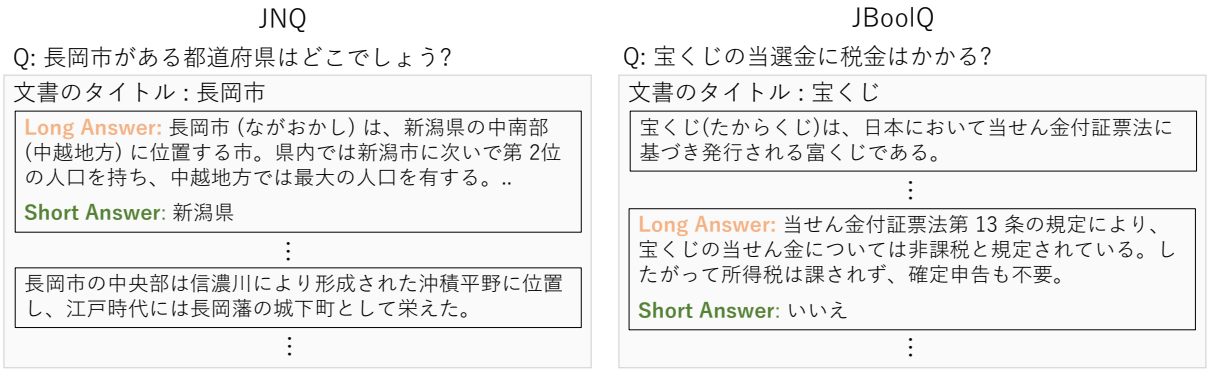

図 1　JNQ と JBoolQ の例

含む段落が複数存在するので、JNQ では 1 つの質問文に対する long answer が複数ある、もしくは 1 つも無い場合を許容する。

図 2 に構築の流れを示す。以下では JNQ 構築の各段階について説明する。クラウドソーシングでは、良質なデータセットを構築するため、1 つの質問に対して 10 人のクラウドワーカーに回答してもらう。

2.1　質問文と文書の収集

JNQ の質問文候補は、Yahoo!検索[2]に入力された検索クエリログから抽出する。人々は検索をする際、文で検索せずに単語を並べて検索する場合がある。このようなクエリは検索エンジンに特化しており、質問文ではないものも含まれているため、スペースを含むクエリは質問文候補から除く。また、短いクエリは、質問形式になっていないことが多いため、8 単語以上で構成されるクエリのみを抽出する[3]。その後、以下のいずれかの質問パターンにマッチしたクエリを抽出する。

1.「は＋疑問詞」を含む
2. 最後の文字が「?」
3.「方法」、「理由」などの特定の単語を含む

得られた質問文候補で Google 検索を行い、上位 5 件以内に日本語 Wikipedia の記事がある場合、最も上位の記事を文書として採用する[4]。

2.2　良い質問文の識別

質問文候補の中には、質問ではない文や不適切な質問文が存在するため、クラウドソーシングを用いて良い質問文を得る。良い質問文は、事実、方法、原因・理由について尋ねる質問と定義し、悪い質問文は、曖昧、前提が誤っている、意見を求める、作品のタイトル、答えるタイミングによって答えが変化する質問と定義する。10 人のクラウドワーカーに質問文候補を与え、良い質問文かどうかを判断してもらう。6 人以上が良いと判断した質問文候補を質問文として採用する。良い質問文と判断された例を付録 A に示す[5]。JNQ から 100 件の質問文を抽出し、英語に翻訳したときにどの wh-word で始まるかに応じて分類した結果を表 1 に示す。

表 1　JNQ の質問文のタイプ分類

タイプ	例	割合
What	ドナウ川が最終的に注いでいる海は?	39%
Where	ナスカの地上絵がある所はどこ?	12%
When	東大寺の大仏はいつ作られた	4%
Why	日本にはなぜ四季があるのか	4%
Who	絵画『ゲルニカ』の作者は誰	3%
How	水道管を凍結させない方法	31%
Yes/No	源泉徴収票は市役所でもらえる?	3%
Other	冬に卵を生で食べられる期間は何日	4%

2.3　Long Answer 抽出

クラウドソーシングを利用して、答えを導くために十分な情報を含む段落である long answer を得る。アノテーションコストを下げるため、クラウドワーカーには最大で 5 段落を与える。この 5 段落は、文書の最初の段落と、2.1 節で行った Google 検索によって得られるスニペットとの関連度が高い上位 4 段落（最初の段落以外）から構成する。これは、概要が述べられることの多い最初の段落と、スニペットとの関連度[6]が高い段落には、答えが含まれる可能性が高いと考えられるからである。この 5 段落に含まれない段落は、long answer ではない段落と判断し、NONE ラベルを付与する（図 2 の点線矢印）。

2）https://search.yahoo.co.jp/
3）単語分割は形態素解析システム Juman++で行う。
4）検索結果の上位 5 件以内に Wikipedia の記事がない質問文候補は除去する。
5）悪いと判断された質問文には、「今日はどこに行こうかな?」、「amazon 支払い方法が承認されません」などがある。
6）関連度は、スニペットと段落をそれぞれ bag of words で表現し、それらの cos 類似度で計算する。

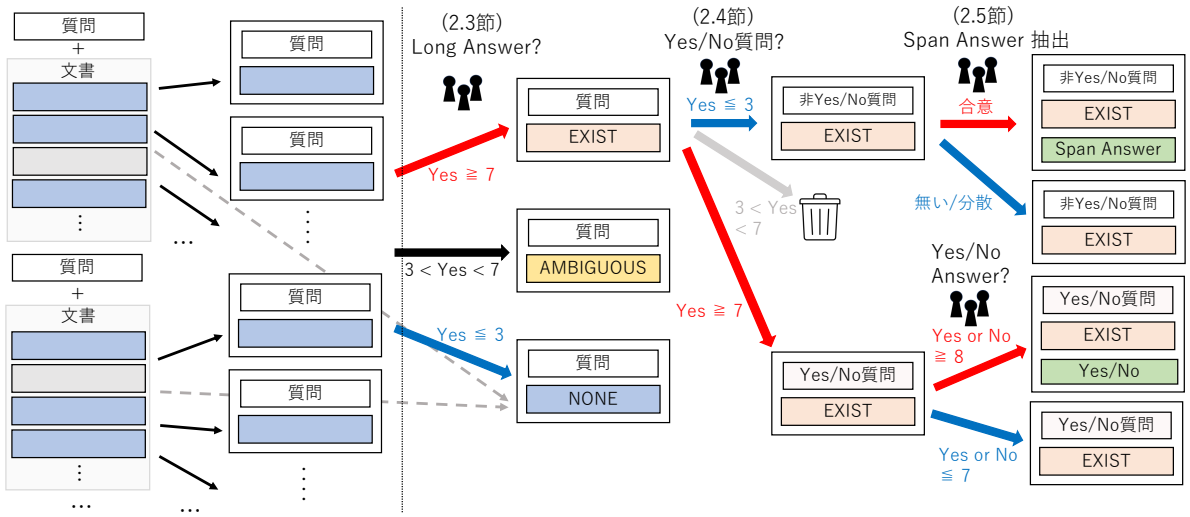

図 2　日本語 Natural Questions の構築フロー

10 人のクラウドワーカーに質問文と各段落を与え、段落が質問の答えを推論するのに十分な情報を含むかどうかを尋ねる。10 人中の票数によって、段落を次の 3 つに分類する。7 人以上が「含む」と回答した場合、その段落を long answer とし、EXIST ラベルを付与する。4 人から 6 人の場合、long answer かどうか曖昧な段落とし、AMBIGUOUS ラベルを付与する。この段落は、学習時に除去することで、ノイズを減らすことができると考えられる。3 人以下の場合は、long answer ではない段落とし、NONE ラベルを付与する。段落ごとに判定するため、1 つの質問に対して複数の long answer が存在する、もしくは、long answer が 1 つも存在しない可能性がある。

2.4　Yes/No 質問識別

次のステップ (2.5 節) では、long answer と判断された段落から short answer を得る。short answer の抽出方法は、質問文が yes/no 質問かどうかで異なるため、質問文が yes/no 質問かどうかをクラウドソーシングで識別する。10 人のうち 7 人以上が yes/no 質問と判断した質問文を yes/no 質問とする。4 人から 6 人が yes/no 質問と判断した場合は、yes/no 質問かどうか曖昧なためデータセットから除去する。

2.5　Short Answer 識別・抽出

質問文が yes/no 質問かどうかによって場合分けし、long answer と判断された段落から short answer を得る。yes/no 質問であれば yes/no answer、yes/no 質問でなければ span answer を得る。

Yes/No Answer 識別　yes/no 質問の場合、答えが「yes」か「no」かを判断してもらう。10 人のクラウドワーカーのうち、8 人以上が「yes」もしくは「no」と判断した場合、その回答を short answer とする。7 人以下の段落は、「yes」か「no」かが曖昧な質問と判断し、short answer には NONE ラベルを付与する。つまり、この段落は long answer のみと判断される。

Span Answer 抽出　yes/no 質問ではない場合、段落から span answer をクラウドワーカーに抜き出してもらう。10 人の回答を集計し、多数決で判定する。ただし、1 票しか獲得していない回答は信頼性が低いとみなし、採用しない。もし、段落内に span answer がない場合、その段落を NONE とする。

3　日本語 BoolQ

BoolQ [11] は、自然な yes/no 質問に焦点を当てた QA データセットである。non-factoid な質問文が多く含まれており、解くために多様な推論能力が必要とされる。各事例は質問文、段落 (NQ における long answer に相当)、答え (yes/no) で構成される。BoolQ は NQ より仕様を簡単化しており、yes/no のどちらかの答えをもつ質問文のみを採用し、文書全体ではなく 1 つの段落を質問文とペアにしている。

日本語 BoolQ (JBoolQ) は、JNQ における yes/no 質問と同じく、質問文、文書、long answer、yes/no answer で構成する。BoolQ とは異なり、各質問は複数の long answer をもつ可能性があり、答えは yes/no 以外に「答えられない」(NONE) を含む。そのため、BoolQ よりも難易度が高く、この問題を解くためには、文書に関するより深い理解が求められる。

表 2 3 つのタスクの統計: Long answer 抽出においては問題数、その他のタスクにおいてはインスタンス数である。

タスク	Train	Dev	Test
Long answer 抽出	13,496	1,687	1,688
Short answer 抽出	6,158	789	761
Yes/No answer 識別	22,357	2,791	2,806

表 3 Long answer 抽出の精度

モデル	Dev			Test		
	P	R	F1	P	R	F1
Tohoku-BERT-base	53.1	67.4	59.4	51.2	68.3	58.5
Tohoku-BERT-large	53.9	67.5	59.9	56.8	66.2	61.2
Waseda-RoBERTa-base	**63.7**	**73.0**	**68.0**	**64.2**	**73.4**	**68.5**
Waseda-RoBERTa-large	57.9	51.4	54.5	57.9	48.3	52.7
人間	-	-	-	46.3	75.8	57.5

JBoolQ は基本的には JNQ と同じ手続きで構築する。JNQ に含まれる yes/no 質問は約 1%と少ないことを鑑み、JNQ よりも多くのクエリログから収集する。各ステップの詳細は 2 節を参照されたい。

4 実験

実験設定 JNQ から 2 つ、JBoolQ から 1 つのタスクを定義する (各タスクの統計を表 2 に示す)。

- Long answer 抽出: 文書中から long answer を持つすべての段落を選択する。
- Short answer 抽出: 段落内から short answer を抽出する[7)]。
- Yes/No answer 識別: 段落を読み、質問に対する答えを「yes」、「no」、「NONE」のいずれかに分類する。

各タスクで、ベースラインとなるモデルを構築し評価する。Long answer 抽出タスクは、各段落が long answer かどうかの 2 値分類として解く。この時、クラウドワーカーに与えた 5 段落以外の段落を long answer ではない段落として学習する。評価尺度は P, R, F1 を用いる。Short answer 抽出タスクは、SQuAD 2.0 [12] に従って解く。評価尺度は EM (Exact Match) と文字単位の F1 を用いる。Yes/No answer 識別タスクでは、各段落が「yes」、「no」、「NONE」のいずれかである 3 値分類として解く。評価尺度は「yes」と「no」についての P, R, F1 のマイクロ平均を用いる。ベースモデルとして、日本語 BERT [13] と RoBERTa [14] を用いる[8)]。

結果 各タスクの評価結果を表 3、4、5 に示す。

7) long answer が存在するとラベル付けされた質問と段落のペアのみを対象とし、yes/no 質問は NQ に従い除去する。

8) Hugging Face 社の Transformers (https://github.com/huggingface/transformers) を用いた。

表 4 Short answer 抽出の精度

モデル	Dev		Test	
	EM	F1	EM	F1
Tohoku-BERT-base	23.3	33.4	23.1	31.3
Tohoku-BERT-large	23.1	32.9	23.3	31.0
Waseda-RoBERTa-base	41.1	49.9	41.7	50.1
Waseda-RoBERTa-large	**45.5**	**53.4**	**45.7**	**53.9**
人間	-	-	51.1	62.5

表 5 Yes/No answer 識別の精度

モデル	Dev			Test		
	P	R	F1	P	R	F1
Tohoku-BERT-base	63.4	**59.6**	61.4	62.5	52.5	57.0
Tohoku-BERT-large	66.0	54.1	59.5	65.1	50.6	56.9
Waseda-RoBERTa-base	58.1	56.8	57.5	59.5	56.2	57.8
Waseda-RoBERTa-large	**68.4**	57.9	**62.7**	**65.5**	**57.4**	**61.2**
人間	-	-	-	75.8	73.0	74.4

人間のスコアは、クラウドソーシングを利用して算出した。ただし、Long answer 抽出タスクの場合のみ、テストセットから 100 問を抽出した。

Long answer 抽出タスクにおいては、Waseda-RoBERTa-base の性能が最も良い。将来的には、本研究で扱った 5 段落以外において、モデルが long answer と判定した段落をクラウドワーカーに提示し、long answer かどうかを判断してもらうということで、データセットの質が上がると考えられる。

Short answer 抽出タスクにおいては、Waseda-RoBERTa-base と Waseda-RoBERTa-large の性能が良く、スコアは人間と近い。

Yes/No answer 識別タスクでは、precision と比較すると recall が低い値を示しており、「yes」か「no」が答えの質問を「NONE」と予測しているケースがあることがわかる。「NONE」ラベルを追加したことで BoolQ よりも難易度が高いと考えられる。

5 おわりに

本研究では、日本語 Natural Questions (JNQ) と日本語 BoolQ (JBoolQ) の 2 つの QA データセットを構築した。質問文は、検索エンジンのクエリログから収集しており、人間の情報欲求に由来する自然なものである。アノテーションは、コストを低減するためにクラウドソーシングで行った。JNQ から Long answer 抽出、Short answer 抽出、JBoolQ から Yes/No answer 識別の合計 3 タスクを定義し、ベースラインモデルの性能を評価した。

構築したデータセットは QA モデルや NLP モデルの訓練、評価、分析に活用でき、日本語においてこれらの研究が促進されることが期待される。

謝辞

本研究は LINE ヤフー株式会社と早稲田大学の共同研究により実施した。

参考文献

[1] 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 **Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing**, pp. 2369–2380, Brussels, Belgium, October-November 2018. Association for Computational Linguistics.

[2] Dheeru Dua, Yizhong Wang, Pradeep Dasigi, Gabriel Stanovsky, Sameer Singh, and Matt Gardner. DROP: A reading comprehension benchmark requiring discrete reasoning over paragraphs. In **Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies**, pp. 2368–2378, Minneapolis, Minnesota, June 2019. Association for Computational Linguistics.

[3] Daniel Khashabi, Sewon Min, Tushar Khot, Ashish Sabharwal, Oyvind Tafjord, Peter Clark, and Hannaneh Hajishirzi. UNIFIEDQA: Crossing format boundaries with a single QA system. In **Findings of the Association for Computational Linguistics: EMNLP 2020**, pp. 1896–1907, Online, November 2020. Association for Computational Linguistics.

[4] Jason Wei, Maarten Bosma, Vincent Zhao, Kelvin Guu, Adams Wei Yu, Brian Lester, Nan Du, Andrew M. Dai, and Quoc V Le. Finetuned language models are zero-shot learners. In **International Conference on Learning Representations**, 2022.

[5] Kentaro Kurihara, Daisuke Kawahara, and Tomohide Shibata. JGLUE: Japanese general language understanding evaluation. In **Proceedings of the Thirteenth Language Resources and Evaluation Conference**, pp. 2957–2966, Marseille, France, June 2022. European Language Resources Association.

[6] ByungHoon So, Kyuhong Byun, Kyungwon Kang, and Seongjin Cho. JaQuAD: Japanese Question Answering Dataset for Machine Reading Comprehension, 2022.

[7] 鈴木正敏, 鈴木潤, 松田耕史, 西田京介, 井之上直也. JAQKET: クイズを題材にした日本語 QA データセットの構築. 言語処理学会第 26 回年次大会 (NLP2020) 発表論文集, pp. 237–240, Online, March 2020. 言語処理学会.

[8] 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.

[9] 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.

[10] Suchin Gururangan, Swabha Swayamdipta, Omer Levy, Roy Schwartz, Samuel Bowman, and Noah A. Smith. Annotation artifacts in natural language inference data. In **Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies**, pp. 107–112, New Orleans, Louisiana, June 2018. Association for Computational Linguistics.

[11] Christopher Clark, Kenton Lee, Ming-Wei Chang, Tom Kwiatkowski, Michael Collins, and Kristina Toutanova. BoolQ: Exploring the surprising difficulty of natural yes/no questions. In **Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies**, pp. 2924–2936, Minneapolis, Minnesota, June 2019. Association for Computational Linguistics.

[12] Pranav Rajpurkar, Robin Jia, and Percy Liang. Know what you don't know: Unanswerable questions for SQuAD. In **Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)**, pp. 784–789, Melbourne, Australia, July 2018. Association for Computational Linguistics.

[13] 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, Volume 1 (Long and Short Papers)**, pp. 4171–4186, Minneapolis, Minnesota, June 2019. Association for Computational Linguistics.

[14] Yinhan Liu, Myle Ott, Naman Goyal, Jingfei Du, Mandar Joshi, Danqi Chen, Omer Levy, Mike Lewis, Luke Zettlemoyer, and Veselin Stoyanov. Roberta: A robustly optimized bert pretraining approach, 2019.

[15] Vladimir Karpukhin, Barlas Oguz, Sewon Min, Patrick Lewis, Ledell Wu, Sergey Edunov, Danqi Chen, and Wen-tau Yih. Dense passage retrieval for open-domain question answering. In **Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)**, pp. 6769–6781, Online, November 2020. Association for Computational Linguistics.

[16] Xilun Chen, Kushal Lakhotia, Barlas Oguz, Anchit Gupta, Patrick Lewis, Stan Peshterliev, Yashar Mehdad, Sonal Gupta, and Wen-tau Yih. Salient phrase aware dense retrieval: Can a dense retriever imitate a sparse one? In Yoav Goldberg, Zornitsa Kozareva, and Yue Zhang, editors, **Findings of the Association for Computational Linguistics: EMNLP 2022**, pp. 250–262, Abu Dhabi, United Arab Emirates, December 2022. Association for Computational Linguistics.

A　良い質問文の例

2.2 節で得られた良い質問文の例を表 6 に示す。

表 6　良い質問文の例

タイプ	例
事実	「九谷焼」はどこの都道府県の工芸品でしょう?
理由	ビール瓶の色が茶色なのはなぜでしょう?
方法	大根おろしの辛味をとる方法

B　各データセットの質問文と段落の数と長さ

JNQ と JBoolQ における質問文と段落数に関する統計情報をそれぞれ表 7、8 に示す。

表 7　JNQ における質問文と段落、span answer の数と長さ。この表の段落は、ラベル付けをしていない段落（つまり long answer ではないと判断される）を含むすべての段落を指す。

	数	長さ (文字数)		
		平均	最大	最小
質問文	16,871	17.8	50	8
段落	192,514	159.0	999	10
Span answer	5,463	9.6	180	1

表 8　JBoolQ における質問文と段落の数と長さ

	数	長さ (文字数)		
		平均	最大	最小
質問文	6,467	11.4	48	6
段落	88,366	154.0	988	10

C　質問タイプ

JBoolQ において 100 件の質問文を抽出し、タイプ別に分類した。その結果を表 9 に示す。

D　Open-Domain タスク

JNQ から Open-Domain タスクを定義し、ベースラインモデルを評価した。Open-Domain タスクは、EfficientQA[9] に従い、文書を参照せずに short answer を答えるタスクである。モデルは TF-IDF(retriever) と DPR reader の組み合わせと、DPR [15] を用いる。統計情報については表 10 に示す。

結果を表 11 に示す。Open-Domain NQ タスクでは、テストにおいて、TF-IDF + DPR reader が DPR よりもわずかに良い結果を示した。質問文の平均長が比較的短く、質問文中の salient なフレーズや希少なエンティティが DPR の正確な検索を難しくしていると推測される [16]。また、「男の子の髪の毛の切り方」のように標準的な答えがなく、open-domain QA に適さない質問文が含まれており、このような質問文は今後除去していく予定である。

9）https://efficientqa.github.io/

表 9　JBoolQ の質問文のタイプ分類

タイプ	例	割合
可能性	新幹線で携帯充電できる?	23%
必要性	履歴書に印鑑は必要か	11%
定義	ナショナルとパナソニックは同じ?	7%
存在	国会議事堂の中に保育園ある?	4%
その他事実 (一般)	疲れで熱は出る?	24%
その他事実 (実体)	久能山東照宮は神社?	31%

表 10　Open-Domain タスクの統計（インスタンス数）

Task	Train	Dev	Test
Open-Domain NQ	2,286	306	306

表 11　Open-domain タスクの精度

	Dev	Test
	EM	
TF-IDF + DPR reader	27.4	29.0
DPR	27.7	26.8