

# XOR QA: Cross-lingual Open-Retrieval Question Answering

Akari Asai<sup>✳</sup>, Jungo Kasai<sup>✳</sup>, Jonathan H. Clark<sup>✳</sup>,  
Kenton Lee<sup>✳</sup>, Eunsol Choi<sup>♥</sup>, Hannaneh Hajishirzi<sup>✳✳</sup>

<sup>✳</sup>University of Washington <sup>✳</sup>Google Research

<sup>♥</sup>The University of Texas at Austin <sup>✳</sup>Allen Institute for AI

{akari, jkasai, hannaneh}@cs.washington.edu

{jhclark, kentonl}@google.com, eunsol@cs.utexas.edu

## Abstract

Multilingual question answering tasks typically assume that answers exist in the same language as the question. Yet in practice, many languages face both *information scarcity*—where languages have few reference articles—and *information asymmetry*—where questions reference concepts from other cultures. This work extends open-retrieval question answering to a cross-lingual setting enabling questions from one language to be answered via answer content from another language. We construct a large-scale dataset built on 40K information-seeking questions across 7 diverse non-English languages that TYDI QA could not find same-language answers for. Based on this dataset, we introduce a task framework, called **Cross-lingual Open-Retrieval Question Answering (XOR QA)**, that consists of three new tasks involving cross-lingual document retrieval from multilingual and English resources. We establish baselines with state-of-the-art machine translation systems and cross-lingual pretrained models. Experimental results suggest that XOR QA is a challenging task that will facilitate the development of novel techniques for multilingual question answering. Our data and code are available at <https://nlp.cs.washington.edu/xorqa/>.

## 1 Introduction

Information-seeking questions—questions from people who are actually looking for an answer—have been increasingly studied in question answering (QA) research. Fulfilling these information needs has led the research community to look further for answers: beyond paragraphs and articles toward performing **open retrieval**<sup>1</sup> on large-scale document collections (Chen and Yih, 2020). Yet

<sup>1</sup>We use **open retrieval**—instead of **open domain**—to refer to models that can access answer context from large document collections. We avoid using open domain due to its double meaning as “covering topics from many domains.”

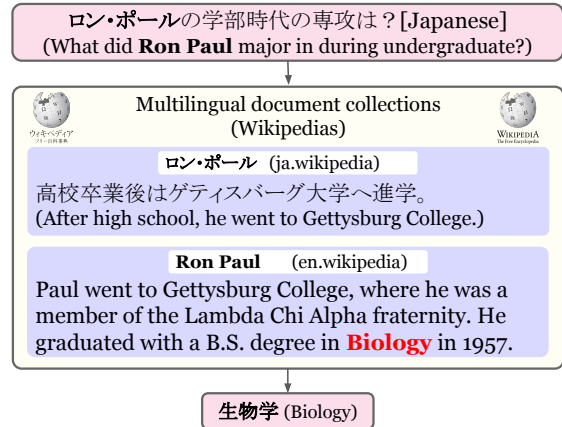


Figure 1: Overview of XOR QA. Given a question in  $L_i$ , the model finds an answer in either English or  $L_i$  Wikipedia and returns an answer in English or  $L_i$ .  $L_i$  is one of the 7 typologically diverse languages.

the bulk of this work has been exclusively on English. In this paper, we bring together for the first time information-seeking questions, open-retrieval QA, and multilingual QA to create a multilingual open-retrieval QA dataset that enables *cross-lingual* answer retrieval.

While multilingual open QA systems would benefit the many speakers of non-English languages, there are several pitfalls in designing such a dataset. First, a multilingual QA dataset should include questions from non-English native speakers to represent real-world applications. Questions in most recent multilingual QA datasets (Lewis et al., 2020; Artetxe et al., 2020; Longpre et al., 2020) are translated from English, which leads to English-centric questions such as questions about American sports, cultures and politics. Second, it is important to support retrieving answers in languages other than the original language due to *information scarcity* of low-resource languages (Miniwatts Marketing Group, 2011). Moreover, questions strongly related to entities from other cultures are less likely to have answer content in the questioner’s language

due to cultural bias (*information asymmetry*, Callahan and Herring, 2011). For example, Fig. 1 shows that the Japanese Wikipedia article of an American politician, Ron Paul, does not have information about his college degree perhaps because Japanese Wikipedia editors are less interested in specific educational backgrounds of American politicians.

In this paper, we introduce the task of cross-lingual open-retrieval question answering (XOR QA) which aims at answering multilingual questions from non-English native speakers given multilingual resources. To support research in this area, we construct a dataset (called XOR-TyDi QA) of 40k annotated questions and answers across 7 typologically diverse languages. Questions in our dataset are inherited from TyDi QA (Clark et al., 2020), which are written by native speakers and are originally unanswerable due to the information scarcity or asymmetry issues. XOR-TyDi QA is the first large-scale cross-lingual open-retrieval QA dataset that consists of information-seeking questions from native speakers and multilingual reference documents.

XOR-TyDi QA is constructed with an annotation pipeline that allows for cross-lingual retrieval from large-scale Wikipedia corpora (§2). Unanswerable questions in TyDi QA are first translated into English by professional translators. Then, annotators find answers to translated queries given English Wikipedia using our new model-in-the-loop annotation framework that reduces annotation errors. Finally, answers are verified and translated back to the target languages.

Building on the dataset, we introduce three new tasks in the order of increasing complexity (§3). In XOR-RETRIEVE, a system retrieves English Wikipedia paragraphs with sufficient information to answer the question posed in the target language. XOR-ENGLISHSPAN takes one step further and finds a minimal answer span from the retrieved English paragraphs. Finally, XOR-FULL expects a system to generate an answer end to end in the target language by consulting both English and the target language’s Wikipedia. XOR-FULL is our ultimate goal, and the first two tasks enable researchers to diagnose where their models fail and develop under less coding efforts and resources.

We provide baselines that extend state-of-the-art open-retrieval QA systems (Asai et al., 2020; Karpukhin et al., 2020) to our multilingual retrieval setting. Our best baseline achieves an average of

18.7 F1 points on XOR-FULL. This result indicates that XOR-TyDi QA poses unique challenges to tackle toward building a real-world open-retrieval QA system for diverse languages. We expect that our dataset opens up new challenges to make progress in multilingual representation learning.

## 2 The XOR-TyDi QA Dataset

Our XOR-TyDi QA dataset comprises questions inherited from TyDi QA (Clark et al., 2020) and answers augmented with our annotation process across 7 typologically diverse languages. We focus on cross-lingual retrieval from English Wikipedia because in our preliminary investigation we were able to find answers to a majority of the questions from resource-rich English Wikipedia, and native speakers with much annotation experience were readily available via crowdsourcing in English.

### 2.1 XOR-TyDi QA Collection

Our annotation pipeline proceeds with four steps: 1) collection of questions from TyDi QA without a same-language answer which require cross-lingual reference to answer (§2.1.1); 2) question translation from a target language to the pivot language of English where the missing information may exist (§2.1.2); 3) answer retrieval in the pivot language given a set of candidate documents (§2.1.3); 4) answer verification and translation from the pivot language back to the original language (§2.1.4). Fig. 2 shows an overview of the pipeline.

#### 2.1.1 Question Selection

Our questions are collected from *unanswerable* questions in TyDi QA. A question is unanswerable in TyDi QA if an annotator cannot select a passage answer (a paragraph in the article that contains an answer). We randomly sample 5,000 questions without any passage answer annotations (unanswerable questions) from the TyDi QA training data, and split them into training (4,500) and development (500) sets. We use the development data from TyDi QA as our test data, since the TyDi QA’s original test data is not publicly available.<sup>2</sup> We choose 7 languages with varying amounts of Wikipedia data out of the 10 non-English languages based on the cost and availability

<sup>2</sup>Furthermore, despite the benefits of hidden test sets, the resource-intensive nature of open-retrieval QA is not suitable to code-submission leaderboards. This further precluded the use of the original TyDi QA test sets.

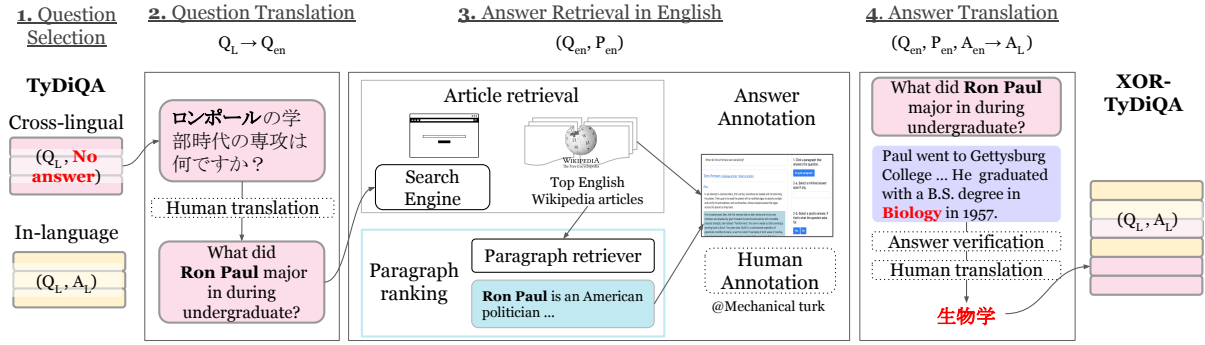


Figure 2: Overview of the annotation process for XOR-TyDi QA.

of translators:<sup>3</sup> Arabic, Bengali, Finnish, Japanese, Korean, Russian and Telugu.

### 2.1.2 Question Translation

We use a professional translation service, Gengo,<sup>4</sup> to translate all collected questions into English. Since named entities are crucial for QA, we instruct translators to carefully translate them by searching for common English translations from English Wikipedia or other external sources. We perform manual quality assessment by native speakers on 50 translation samples, finding that more than 95% are correct. Note that while these translations are a part of the annotation procedure (due to the inherently cross-lingual nature of this task), they are *not* provided to models during evaluation.

### 2.1.3 Answer Retrieval in English

We use Amazon Mechanical Turk to retrieve answers to translated English questions given English Wikipedia articles. Annotators are instructed to select passage answers (gold paragraphs) and minimal answer spans as in Clark et al. (2020).

To annotate answers to information-seeking queries, previous work first identifies relevant Wikipedia articles using Google Search, and then annotators attempt to find answers there. Asai and Choi (2020) show that in information-seeking QA datasets many questions were annotated as “unanswerable” due to two systematic errors: *retrieval error* where the search engine failed to retrieve a relevant article and *answer annotation error* where the annotator overlooks answer content. Importantly, these two types of annotation errors present a tradeoff: if we retrieve many articles, retrieval errors will be reduced at the expense of answer

annotation errors because annotators have to find answer context among many candidate articles.

**Collaborative model-in-the-loop.** To find a middle ground in the tradeoff, we introduce a collaborative model-in-the-loop framework that uses Google Search and a state-of-the-art paragraph ranker. We first run Google Search to retrieve as many as top 10 Wikipedia articles, resulting in 387 paragraphs per question on average. We score them with Path Retriever (Asai et al., 2020) and present the five highest scoring paragraphs. Annotators are asked to skim these five paragraphs first; if they cannot find any answer content, they are asked to read the rest of the paragraphs, where the Wikipedia section headings guide their reading. To incentivize workers to find answers beyond the pre-selected ones, we carefully communicate with workers and send additional rewards to annotators who actively read the rest of the paragraphs and find answers for questions that other annotators may overlook. We found about 70% of the answers from the 5 paragraphs and 30% from the rest of the paragraphs in the top 10 articles. This means that while our paragraph ranking was effective, the annotators did not fully rely on it, thereby mitigating the influence of the passage ranking model on the dataset. See Appendix §B.1 for annotation interface details.

**Quality control for QA annotation.** We first recruit MTurkers with a high approval rate ( $\geq 96\%$ ) located in English-speaking countries, and all workers first annotate the same qualification batch. We assess the quality of those submissions and select high-quality annotators. Consequently, 40 out of more than 200 workers were qualified and 24 workers annotated most of our data. More details are in Appendix B.3.

<sup>3</sup>The cost of translations depends on the number of available translators, and the estimated translation cost for the other three non-English languages was considerably higher.

<sup>4</sup><https://gengo.com/>.

%	Ar	Bn	Fi	Ja	Ko	Ru	Te	All
TYDI QA	82	42	57	50	29	69	28	50
XOR-TYDI QA	92	82	83	77	68	83	44	72
Improvement	10	40	26	27	39	14	16	22

Table 1: Percentage of the questions with short answers (answerable questions) in the original TYDI QA dataset (dev) and XOR-TYDI QA. The third row (Improvement) represents the percentage of the questions that become answerable by searching the English Wikipedia articles.

#### 2.1.4 Answer Verification and Translation

We verify the annotated answers and translate those answers back to the target languages (cross-lingual data). Finally, we mix the annotated cross-lingual data with the same-language data from TYDI QA to reflect the actual question distributions from native speakers (in-language data).

**Answer verification.** We trained undergraduate students who are native English speakers to verify the annotated paragraphs and short answers. Only 8% of the answers were marked as incorrect through the verification phase and were later corrected by our pool of high-quality crowdworkers who yielded less than 1% annotation error.

**Answer translation.** We again use Gengo to translate answers from English back to the original languages. We give translators further instructions to normalize answers such that they are consistent with answers in TYDI QA. For example, some languages use their own unique set of numerals rather than Arabic numerals to represent numeric answers (e.g., Bengali numerals, Chinese numerals in Japanese text). The details of the answer translation process are described in Appendix §B.4. Note that because of the cost of answer translations, we conduct this answer translation process for evaluation sets only.

## 2.2 The XOR-TYDI QA Corpus

**Dataset statistics.**<sup>5</sup> Table 1 shows the percentages of the questions annotated with short answers in the original TYDI QA and our XOR-TYDI QA, and Table 2 shows statistics of XOR-TYDI QA. As seen in Table 1, cross-lingual retrieval significantly increases the answer coverage in all languages by up to 40% (Bengali), and consequently we found answers for more than 50% of the origi-

<sup>5</sup>After our initial release in November 2020, we modified the XOR-TYDI QA data, and released a new version as XOR-TYDI QA (v1.1). All results are based on v1.1.

	Cross-lingual			In-language		
	Train	Dev	Test	Train	Dev	Test
Ar	2,574	350	137	15,828	358	1,132
Bn	2,582	312	128	2,428	115	139
Fi	2,088	360	530	7,680	255	1,197
Ja	2,288	296	449	5,527	137	867
Ko	2,469	299	646	1,856	72	505
Ru	1,941	255	235	7,349	313	1,125
Te	1,308	238	374	5,451	113	712

Table 2: Dataset size of the XOR-TYDI QA corpus (answered data). **Cross-lingual** data comes from our re-annotated questions that did not originally have same-language answers in TYDI QA. **In-language** data are taken directly from answerable questions in TYDI QA.

nal information-seeking questions in 6 out of the 7 languages.<sup>6</sup> This result confirms the effectiveness of searching multilingual document collections to improve the answer coverage. Detailed statistics of the numbers of long answers, short answers, and unanswered questions are in Appendix §B.5. We also release the 30k manually translated questions for our training set, which could be used to train multilingual models or machine translation models.

**Qualitative examples.** Table 3 illustrates that finding relevant articles from multilingual document collections is important to answer questions asked by users with diverse linguistic and cultural backgrounds. The first question is unanswerable in Korean Wikipedia, but there is a clear description about who was the prime minister of France at the time in English Wikipedia. The second example shows English Wikipedia sometimes contains rich information about a target language-specific topic (e.g., economy in Krasnodar, a city in Russia). Those examples demonstrate the effectiveness of searching for answers in another language with more abundant knowledge sources. In the last question of Table 3, on the other hand, only the Wikipedia of the target language can provide the answer. XOR QA allows for both retrieval paths.

**Comparison with other datasets.** Table 4 compares XOR-TYDI QA and existing multilingual QA datasets. XOR-TYDI QA has three key properties that are distinct from these QA benchmarks. First, since all questions are inherited from TYDI QA, they are information-seeking questions written by

<sup>6</sup>We found in the Telugu data, certain types of questions are very frequent (e.g., what is the pin code of X mandal?). Those questions often ask some specific information of local administration districts, and are often unanswerable because (a) they are typically not described in English Wikipedia and (b) the overall coverage of Telugu Wikipedia is quite low.



$L$	Original Question: $Q_L$ ( $Q_{en}$ )	Passage Answer: $P_{en}$ or $P_L$	Minimal Answer in English: $A_{en}$	Final Answer: $A_L$
Ko	1993년 프랑스 총리는 누구 인가요? (Who was the French Prime Minister in 1993?)	Mayor of Neuilly-sur-Seine from 1983 to 2002, he was Minister of the Budget under Prime Minister Édouard Balladur (1993–1995).	Édouard Balladur	에두아르 발라뒤르
Ru	Какая средняя зарплата в Краснодаре на сегодняшний день? (What is the average wage in <b>Krasnodar</b> ?)	Krasnodar has the lowest unemployment rate among the cities of the Southern Federal District at 0.3% of the total working-age population. In addition, Krasnodar holds the first place in terms of highest average salary—21,742 rubles per capita.	21,742 rubles	21,742 рубля
Ja	速水堅曹はどこで製糸技術を学んだ? (Where did <b>Kenso Hayami</b> learn the silk-reeling technique?)	藩営前橋製糸所を前橋に開設。カスバル・ミューラーから直接、器械製糸技術を学び (he founded Hanei Maebashi Silk Mill and learned instrumental silk reeling techniques directly from Caspal Müller)	–	藩営前橋製糸所 (Hanei Maebashi Silk Mill)

Table 3: Examples newly annotated for Korean (Ko) and Russian (Ru) questions. The bottom example is an answerable question from TYDI QA for which only Japanese Wikipedia includes the correct answer.

Dataset	Asked by native speakers	Open-retrieval	Cross-lingual
TYDI QA	✓	✗	✗
MLQA	✗	✗	✓
XQuAD	✗	✗	✗
MKQA	✗	WikiData	✗
MLQA-R	✗	21k sents	✓
XQuAD-R	✗	13k sents	✓
XOR-TYDI QA	✓	Wikipedia	✓

Table 4: Comparison with recent multilingual QA datasets. MKQA’s answers are aligned to WikiData.

native speakers, and better reflect native speakers’ interests and their own linguistic phenomena. This distinguishes XOR-TYDI QA from translation-based datasets such as MLQA (Lewis et al., 2020) and MKQA (Longpre et al., 2020). Second, our dataset requires cross-lingual retrieval unlike other multilingual datasets such as TYDI QA or XQuAD (Artetxe et al., 2020), which focus on same-language QA. Lastly, questions in XOR-TYDI QA require open retrieval from Wikipedia, whereas MLQA-R and XQuAD-R (Roy et al., 2020) limit the search space to matching each question with the predetermined 21k/31k sentences.

### 3 XOR QA Tasks and Baselines

We introduce three new tasks (Fig. 3): **XOR-RETRIEVE**, **XOR-ENGLISHSPAN**, and **XOR-FULL** with our newly collected XOR-TYDI QA dataset and construct strong baselines for each task. XOR-FULL defines our goal of building a multilingual open-retrieval QA system that uses both cross-

lingual and in-language questions from XOR-TYDI QA. To diagnose where models fail and to allow researchers to use the data with less coding effort or computational resource, we also introduce the first two intermediate tasks that only use the cross-lingual data (Table 2). We denote the target language by  $L_i$ . We also denote the English Wikipedia collection by  $W_{eng}$  and the Wikipedia collection in each target language  $L_i$  by  $W_i$ . We experiment with baselines using black-box APIs as a reference, but we encourage the community to use white-box systems so that all experimental details can be understood. Nonetheless, we release the intermediate results from those external APIs to make our results reproducible. All of the white-box system results can be reproduced using our codebase.

#### 3.1 XOR-RETRIEVE: Cross-lingual Paragraph Retrieval

**Task.** Given a question in  $L_i$  and English Wikipedia  $W_{eng}$ , the task is to retrieve English paragraphs for the question. Finding evidence paragraphs from large-scale document collections like Wikipedia is a challenging task, especially when a query and documents are in different languages and systems cannot perform lexical matching.

**Evaluation.** Different open-retrieval QA models use different units for retrieval. To make fair comparisons across various models, we measure the recall by computing the fraction of the questions for which the minimal answer is contained in the top  $n$  tokens selected. We evaluate with  $n = 2k, 5k$ : R@2kt and R@5kt (kilo-tokens).

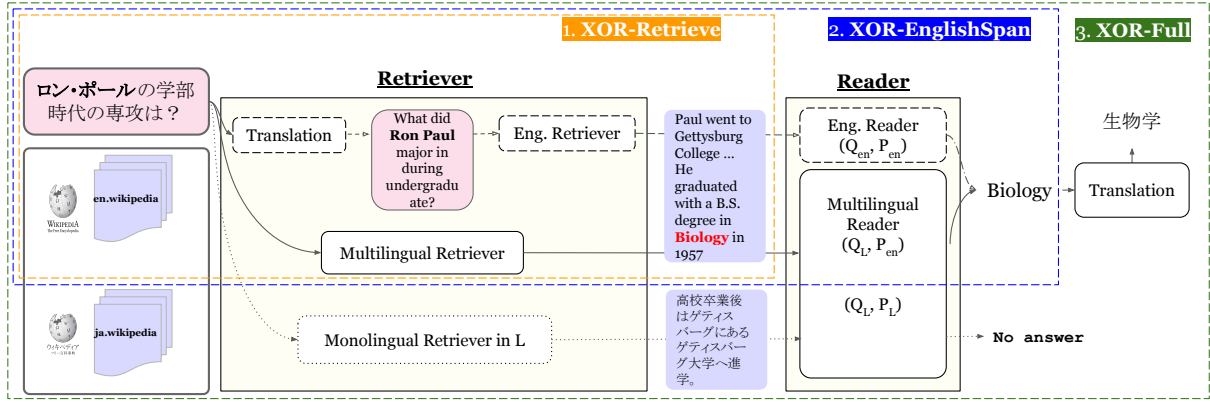


Figure 3: Overview of the tasks and baselines. Each dotted rectangle represents one of the three tasks and surrounds used pipeline modules.

**Translate baselines.** We first translate queries into English, and then paragraphs are retrieved in a monolingual way. For query translation, we train transformer machine translation (MT) models on publicly available corpora for easy replication. We also run Google’s online machine translation service (GMT). This is not completely reproducible as these systems get constantly updated; nor do we know what model and training data they use. We encourage the community to use open MT systems where system details are available. For retrieval, we explore term-based retrieval (BM25, [Robertson and Zaragoza 2009](#)), term-based retrieval followed by neural paragraph ranking (Path Retriever, [Asai et al. 2020](#)), and end-to-end neural retrieval (DPR, [Karpukhin et al. 2020](#)).

**Multilingual baselines.** Alternatively, we can directly apply a multilingual pretrained model to retrieve paragraphs. We initialize and train a DPR encoder with multilingual BERT to enable multilingual document retrieval ([Devlin et al., 2019](#)).

### 3.2 XOR-ENGLISHSPAN: L-to-English Open-Retrieval QA

**Task.** Given a question in  $L_i$  and English Wikipedia  $W_{eng}$ , a system retrieves paragraphs from  $W_{eng}$  and extracts an answer. This task is equivalent to existing open-retrieval QA tasks ([Chen et al., 2017](#)), except that the query is not in English. This task involves challenging cross-lingual retrieval and question answering on the  $L_i$  query and English evidence paragraphs.

**Evaluation.** We use Exact Match (EM) and F1 over the annotated answer’s token set following prior work ([Rajpurkar et al., 2016](#)).

**Baselines.** Our pipeline uses a machine reading

model to find a minimal span that answers the question given paragraphs selected from the previous XOR-RETRIEVE step. In particular, for the translate baselines, we use the same approach as state-of-the-art models ([Asai et al., 2020](#); [Karpukhin et al., 2020](#)) that jointly predicts a span and a relevance score of each paragraph to the question. For the multilingual baseline where queries are *not* automatically translated during evaluation, we build a reader model with multilingual BERT.

### 3.3 XOR-FULL: Round Trip

**Task.** Given a question in target language  $L_i$  and Wikipedia in both English and  $L_i$  ( $W_{eng}$  and  $W_i$ ), a system is required to generate an answer in  $L_i$ . In this task, a system does not know *a priori* in which language we can find information that the user is seeking. Note that the XOR-FULL evaluation data includes both cross-lingual and in-language data, while XOR-RETRIEVE and XOR-ENGLISHSPAN only use cross-lingual data during evaluation.

**Evaluation.** Some answers in XOR-FULL are translated from English so the same spans may not exist in the target language’s Wikipedia. For this reason, we use token-level BLEU scores ([Papineni et al., 2002](#)) over a ground-truth token set in addition to F1 and EM. The same tokenizer is applied to ground-truth and predicted answers to compute token-level F1 and BLEU.<sup>7</sup>

**Baselines.** Unlike the previous two tasks, evidence paragraphs can be found both in the target language and English, and a system has to output final answers based on the most plausible paragraphs. In this work, we introduce a simple multi-

<sup>7</sup>We use the Moses tokenizer ([Koehn et al., 2007](#)) for all languages except we apply MeCab ([Kudo, 2006](#)) to Japanese.

lingual baseline that first looks for answers in the target language and then English if no answers are found in the target language. Specifically, we apply monolingual retrieval (i.e., BM25, Google Custom Search) for  $W_i$  and a multilingual machine reading model based on XLM-RoBERTa (Conneau et al., 2020) to find in-language answers in the target language (monolingual model; the bottom half of Fig. 3). If no answers are found by the monolingual model, we apply an XOR-ENGLISHSPAN baseline and translate English answers into the target language (the top half of Fig. 3).

## 4 Experiments and Analysis

We present results from the baselines discussed above. We find that the three XOR QA tasks present challenges even for the strong models.

### 4.1 Experimental Setup

For training, we first finetune the retrieval and machine reading models with the Natural Questions data (Kwiatkowski et al., 2019) and then further finetune on our XOR-TYDI QA data. For the BM25 retrieval baseline, we use ElasticSearch<sup>8</sup> to store and search documents using BM25 similarities. For both Path Retriever and DPR, we run the official open-source code. For our MT systems, we train base-sized (large for Russian) autoregressive transformers (Vaswani et al., 2017) on parallel corpora from OPUS (Tiedemann and Nygaard, 2004), MultiUN (Ziems et al., 2016), or WMT19 (Barrault et al., 2019). All data are encoded into subwords by BPE (Sennrich et al., 2016) or SentencePiece (Kudo and Richardson, 2018). We use the fairseq library (Ott et al., 2019). Additional experimental details and full lists of hyperparameters are available in Appendix §C.

We only evaluate questions having answers and do not give credit to predicting “no answers” as in prior open-retrieval work (Lee et al., 2019). For XOR-RETRIEVE and XOR-ENGLISHSPAN, we use cross-lingual data only and both cross-lingual and in-language data for XOR-FULL.

### 4.2 XOR-RETRIEVE Experiments

Table 5 shows the R@5kt (as defined in §3.1) for different retrieval and query translation systems.<sup>9</sup> We also report the performance with the human

	Human Translation			GMT		Our MT		<i>Multi.</i> DPR
	DPR	PATH	BM	DPR	PATH	DPR	PATH	
Ar	68.3	<b>70.0</b>	41.6	67.5	63.3	52.5	51.6	50.4
Bn	<b>85.6</b>	82.0	57.0	83.2	78.9	63.2	64.8	57.7
Fi	<b>73.1</b>	70.2	43.7	68.1	64.1	65.9	59.5	58.9
Ja	<b>68.9</b>	63.0	38.8	60.1	52.3	52.1	41.7	37.3
Ko	<b>70.9</b>	63.6	43.8	66.3	54.0	46.5	37.6	42.8
Ru	<b>65.2</b>	63.7	35.2	60.4	56.5	47.3	38.1	44.0
Te	<b>72.2</b>	64.1	44.6	65.0	62.5	22.7	18.1	44.9
Av.	<b>72.1</b>	68.1	43.5	67.2	61.7	50.0	44.5	48.0

Table 5: R@5kt (§3.1) on the test data in the XOR-RETRIEVE setting. PATH and BM denote Path Retriever and BM25 respectively. *Multi.* is a multilingual approach that bypasses the query translation step.

English translations of the questions used during the dataset collection as an upper bound of translate baselines. The best R@5kt macro-averaged over the 7 languages comes from running DPR on human translations: 72.1. Machine translation systems achieve averages of 67.2 (GMT) and 50.0 (our MT) again with DPR. The discrepancy between human and machine translation suggests that even state-of-the-art translation systems struggle to translate questions precisely enough to retrieve an evidence paragraph. Although the difference between GMT and our MT systems shows the effectiveness of industrial MT systems (large parallel data, model architecture, etc.), there remains a substantial performance gap from human translation. The translate baselines outperform the multilingual approach apart from Telugu, where our MT suffers from small parallel data (114k sentences), and as a result the multilingual approach performs better.

BM25 substantially underperforms the other two models across the board. DPR generally achieves similar performance, if not better, compared to Path Retriever despite the fact that Path Retriever was used in our annotation (§2.1.3). As we found that these patterns persisted in all the following experiments, we will only report results with DPR.

### 4.3 XOR-ENGLISHSPAN Experiments

Table 6 shows the performance of the baseline models in XOR-ENGLISHSPAN. The average macro F1 score with queries translated by human translators is 38.2, substantially higher than that of MT-based models: 32.9 and 20.5 F1 points for GMT and our MT respectively. This suggests that errors in automatic query translation affect later layers in the pipeline. The multilingual approach consistently underperforms translation-based methods, similarly to XOR-RETRIEVE. As in XOR-RETRIEVE,

<sup>8</sup><https://www.elastic.co/jp/>.

<sup>9</sup>We measured R@2kt as well (Table 12 in Appendix), but the relative pattern persisted across languages and methods.

	Human Translation		GMT		Our MT		<i>Multi.</i>	
	F1	EM	F1	EM	F1	EM	F1	EM
Ar	<b>43.2</b>	<b>32.8</b>	39.5	28.5	28.0	23.4	17.9	11.7
Bn	<b>43.4</b>	<b>35.9</b>	42.1	34.4	25.6	20.3	19.4	14.1
Fi	<b>34.8</b>	<b>26.0</b>	28.2	21.3	29.3	22.1	24.5	18.3
Ja	<b>29.9</b>	<b>22.3</b>	23.5	17.4	19.2	13.8	13.1	10.7
Ko	<b>36.9</b>	<b>28.8</b>	30.5	23.8	19.4	14.2	14.3	9.9
Ru	<b>37.0</b>	<b>29.4</b>	34.8	26.4	18.4	13.6	17.2	11.1
Te	<b>42.4</b>	<b>35.0</b>	31.6	25.1	3.8	2.7	14.4	10.2
Av.	<b>38.2</b>	<b>30.0</b>	32.9	25.3	20.5	15.7	17.2	12.3

Table 6: Performance on XOR-ENGLISHSPAN. The rightmost *Multi.* section is a multilingual approach without query translation (§3.1).

Telugu was an exception. The multilingual baseline significantly outperforms the translation-based approach with our MT system (14.4 vs. 3.6 F1 points). Query translation errors propagate to and directly impact downstream QA tasks in the languages with limited parallel data for MT training, and machine translation-based approaches may perform poorly. This encourages the research community to explore multilingual pretrained models to build a robust multilingual open-retrieval QA system for low-resource languages.

Similar to the original TYDI QA dataset, the performance on XOR-ENGLISHSPAN varies across languages, which can be partially explained by the differing sets of questions (Clark et al., 2020). The best baseline achieves 39.5 in Arabic compared to 23.5 F1 points in Japanese, which may come from differences in question difficulty as well as how the models are trained for each language.

#### 4.4 XOR-FULL Experiments

Table 7 presents results on the XOR-FULL task. The first pipeline, which uses GMT, Google Search (GS), and DPR, yields the best average performance: 18.7 F1, 12.1 EM, and 16.8 BLEU points. This indicates that systems like GMT and GS, which are typically trained on large data, are effective. Yet, we encourage the community to experiment on top of open systems such that all experimental details can be fully reported and understood. Replacing GMT with our MT (second row) results in a large performance drop in Bengali (6.6 vs. 19.0 F1 points) and Telugu (1.7 vs. 13.6). Further replacing GS with BM25 retrieval in the target languages (third row) causes a large performance drop in all languages (e.g., 9.7 vs. 16.4 in Korean). Consistent with the previous tasks, the multilingual

approach shown in the forth row underperforms the translation-based counterpart (15.7 vs. 18.7 F1 points on average). Similar baselines perform considerably better in prior open-retrieval QA datasets, such as MKQA (30 EM points, Longpre et al., 2020) and NQ questions (40 F1, Karpukhin et al., 2020). This gap illustrates the multidimensional challenge of XOR-TYDI QA.

#### 4.5 Further Analysis

**Effects of translation performance on overall QA results.** Table 8 compares the query translation BLEU scores and the final QA F1 performance of the translation-based baseline with three different MT systems in XOR-ENGLISHSPAN: **GMT**, **Our MT**, and **Helsinki** (Tiedemann and Thottungal, 2020). GMT significantly outperforms the other two baselines, demonstrating that its training setup may yield large improvements in these languages; similarly, in cases where additional parallel training data is not available, multilingual models may remain strong modeling tools. On the other hand, it is noteworthy that high BLEU scores do not always lead to better QA performance. In Bengali and Finnish, while Helsinki achieves a considerably better BLEU score than our MT (33.0 vs. 30.8 in Bengali and 29.8 vs. 27.4 in Finnish), our MT is 3.9 and 1.3 F1 points better in downstream XOR-ENGLISHSPAN, respectively. See Appendix §D.3 for an example of translation errors resulting in QA errors. Those results suggest that the BLEU score is not always indicative of the downstream performance and that evaluating MT performance in the context of XOR QA would be important for improvements of multilingual QA systems.

**Single language Wikipedia ablations in XOR-FULL.** To assess our models’ ability to benefit from multilingual collections, we try restricting the retrieval target to single language Wikipedia: English  $W_{eng}$  only or target language  $W_i$  only. In  $W_{eng}$  only, the best system, which applies GMT and DPR, underperforms the best pipeline that uses both  $W_{i,eng}$  in all languages except for Finnish and Japanese. Similarly, the  $W_i$  only setting generally underperforms the best  $W_{i,eng}$  pipeline. These results illustrate the importance of searching multilingual collections. See Table 15 for the full results.

### 5 Related Work

**Multilingual QA** Much recent effort has been made to create non-English QA datasets to over-



Translation		Retrieval		Target Language $L_i$ F1								Macro Average		
Query	Answer	$L_i$	Eng.	Ar	Bn	Fi	Ja	Ko	Ru	Te		F1	EM	BLEU
GMT	GMT	GS	DPR	<b>31.5</b>	19.0	<b>18.3</b>	<b>8.8</b>	<b>20.1</b>	<b>19.8</b>	<b>13.6</b>		<b>18.7</b>	<b>12.1</b>	<b>16.8</b>
Our MT	Our MT	GS	DPR	29.6	6.6	15.5	7.6	16.4	18.7	1.7		13.7	8.7	12.0
Our MT	Our MT	BM25	DPR	12.1	<b>22.0</b>	9.3	5.4	9.7	7.4	0.8		9.5	6.0	8.9
–	GMT	GS	DPR	30.5	10.6	16.9	8.2	17.6	<b>19.8</b>	6.0		15.7	10.0	13.9

Table 7: Performance on XOR-FULL (test data F1 scores). “GS” denotes Google Search retrieval.

Query Translator	MT BLEU							XOR-ENGLISHSPAN F1						
	Ar	Bn	Fi	Ja	Ko	Ru	Avg	Ar	Bn	Fi	Ja	Ko	Ru	Avg
GMT	53.9	86.9	30.2	38.2	44.7	52.9	51.8	35.4	42.1	31.8	27.2	32.5	34.7	34.0
Our MT	33.7	30.8	27.4	19.7	30.8	21.7	27.4	20.9	25.2	31.9	19.6	25.3	16.1	23.2
Helsinki	35.9	33.0	29.8	19.8	31.8	37.3	31.1	28.4	21.3	30.6	19.0	25.3	29.6	25.7

Table 8: F1 scores on XOR-ENGLISHSPAN and the BLEU scores in query translation on the dev set. All configurations use DPR. Telugu is excluded since Helsinki does not support it as of October, 2020.

come the data scarcity in non-English languages. In addition to the datasets we already discussed in §2.2, several other non-English reading comprehension datasets have been created (Asai et al., 2018; Lim et al., 2019; Mozannar et al., 2019; d’Hoffschmidt et al., 2020). Liu et al. (2019) developed a template-based *cloze* task, leading to different data distributions from realistic questions with a great degree of lexical overlap between questions and reference paragraphs (Lee et al., 2019). More recently, Hardalov et al. (2020) introduced EXAMS, a multilingual multiple-choice reading comprehension dataset from school exams.

Our XOR-TYDI QA is also closely related to QA@CLEF 2003-2008 (Magnini et al., 2003, 2004; Vallin et al., 2005; Magnini et al., 2006; Giampiccolo et al., 2007; Forner et al., 2008); both QA@CLEF and XOR-TYDI QA attempt to develop and evaluate multilingual QA systems. Nevertheless, there are three crucial differences. First, our XOR-TYDI QA has a large number of questions that are required for training current state-of-the-art QA models like DPR, while QA@CLEF only has 200 evaluation questions for each language without training data (Forner et al., 2010). Secondly, the languages tested in QA@CLEF are all European languages, with the one exception of Indonesian; XOR-TYDI QA includes typologically diverse languages. Lastly, the task setup of QA@CLEF 2003-2008 is either monolingual—questions and documents are written in the same non-English language—or cross-lingual—the source and target languages are pre-specified (Forner et al., 2010). In XOR QA, questions are asked in a target language but a system

does not know in which language it can find an answer in a non-parallel Wikipedia collection. Those differences from QA@CLEF tasks better simulate real-world scenarios and introduce new challenges that have yet to be extensively studied.

**Cross-lingual Information Retrieval** Cross-lingual Information Retrieval (CLIR) is the task of retrieving relevant documents when the document collection is in a different language from the query language (Hull and Grefenstette, 1996). The retrieval component in XOR QA is closely related to CLIR, but differs in several critical ways. First, since the end goal of XOR QA is QA, XOR QA queries always take question forms rather than search key words. Further, while CLIR typically retrieves documents from a single (low-resource) language (Zhang et al., 2019), XOR QA considers documents from both English and the query language. In many applications, we do not know *a priori* in which language we can find target information. Lastly, our document collection is orders of magnitude bigger than typical CLIR benchmarks (Sasaki et al., 2018; Zhang et al., 2019).

## 6 Conclusion

We presented the task of XOR QA, in which a system retrieves and reads documents across languages to answer non-English information-seeking questions. We introduced a new large-scale XOR QA dataset, XOR-TYDI QA, with 40k newly annotated open-retrieval questions that cover seven typologically diverse languages. Our experiments showed that XOR-TYDI QA is a challenging benchmark that can benefit from further effort in both QA and multilinguality communities.

## Acknowledgments

This research was supported by gifts from Google, the Allen Distinguished Investigator Award, the Sloan Fellowship, and the Nakajima Foundation Fellowship. We thank Sewon Min, Kristina Toutanova, David Wadden, the members of the UW NLP group, and the anonymous reviewers for their insightful feedback on this paper, Nancy Li, Xun Cao, Hitesh Boinpally, Samek Mulepati, Casey Zhao, Vitaly Nikolaev, Soumyadip Sengupta, Bindita Chaudhuri, and Aditya Kusupati for their help on our annotations and dataset proofing, and Nelson Liu and Pradeep Dasigi for their suggestions on the annotation interface and Amazon Mechanical Turk crowdsourcing.

## Legal and Ethical Considerations

**Were workers told what the dataset would be used for and did they consent?** Crowdworkers consented to have their responses used in this way through the Amazon Mechanical Turk Participation Agreement.

**If it relates to people, could this dataset expose people to harm or legal action?** Our dataset can include incorrect information to the extent that Wikipedia can have wrong information about people. Nonetheless, we performed extensive quality control and answer verification to minimize the risk of harming people.

**If it relates to people, does it unfairly advantage or disadvantage a particular social group?** One fundamental problem with the existing question answering benchmarks is that most of their questions are written by native English speakers and overly represent English-centric topics, such as American politics, sports, and culture. As such, models trained and developed on those datasets are likely to fail to serve people with diverse language and cultural backgrounds. XOR-TYDI QA remedies this long-standing problem by annotating questions from native speakers of diverse languages. Thus, we encourage researchers and developers to benchmark on XOR-TYDI QA to mitigate the potential bias and unfairness of QA systems. We acknowledge, however, that this dataset still covers a very limited subset of languages in the world. We release a datasheet (Geburu et al., 2018) for our dataset to further document ethical implications.<sup>10</sup>

<sup>10</sup>[https://nlp.cs.washington.edu/xorqa/XORQA\\_site/xorqa\\_datasheet.pdf](https://nlp.cs.washington.edu/xorqa/XORQA_site/xorqa_datasheet.pdf).

## References

- Mikel Artetxe, Sebastian Ruder, and Dani Yogatama. 2020. [On the cross-lingual transferability of monolingual representations](#). In *ACL*.
- Akari Asai and Eunsol Choi. 2020. [Challenges in information seeking QA: Unanswerable questions and paragraph retrieval](#).
- Akari Asai, Akiko Eriguchi, Kazuma Hashimoto, and Yoshimasa Tsuruoka. 2018. [Multilingual extractive reading comprehension by runtime machine translation](#).
- Akari Asai, Kazuma Hashimoto, Hannaneh Hajishirzi, Richard Socher, and Caiming Xiong. 2020. [Learning to retrieve reasoning paths over Wikipedia graph for question answering](#). In *ICLR*.
- Loïc Barrault, Ondřej Bojar, Marta R. Costa-jussà, Christian Federmann, Mark Fishel, Yvette Graham, Barry Haddow, Matthias Huck, Philipp Koehn, Shervin Malmasi, Christof Monz, Mathias Müller, Santanu Pal, Matt Post, and Marcos Zampieri. 2019. [Findings of the 2019 conference on machine translation \(WMT19\)](#). In *WMT*.
- Ewa S Callahan and Susan C Herring. 2011. [Cultural bias in Wikipedia content on famous persons](#). *JASIST*.
- Danqi Chen, Adam Fisch, Jason Weston, and Antoine Bordes. 2017. [Reading Wikipedia to answer open-domain questions](#). In *ACL*.
- Danqi Chen and Wen-tau Yih. 2020. [Open-domain question answering](#). In *ACL: Tutorial Abstracts*.
- Jonathan H. Clark, Eunsol Choi, Michael Collins, Dan Garrette, Tom Kwiatkowski, Vitaly Nikolaev, and Jennimaria Palomaki. 2020. [TyDi QA: A benchmark for information-seeking question answering in typologically diverse languages](#). *TACL*.
- Alexis Conneau, Kartikay Khandelwal, Naman Goyal, Vishrav Chaudhary, Guillaume Wenzek, Francisco Guzmán, Edouard Grave, Myle Ott, Luke Zettlemoyer, and Veselin Stoyanov. 2020. [Unsupervised cross-lingual representation learning at scale](#). In *ACL*.
- Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. [BERT: Pre-training of deep bidirectional transformers for language understanding](#). In *NAACL*.
- Martin d’Hoffschmidt, Wacim Belblidia, Quentin Heinrich, Tom Brendlé, and Maxime Vidal. 2020. [FQuAD: French question answering dataset](#). In *Findings of EMNLP*.
- Pamela Forner, Danilo Giampiccolo, Bernardo Magnini, Anselmo Peñas, Álvaro Rodrigo, and Richard Sutcliffe. 2010. [Evaluating multilingual question answering systems at CLEF](#). In *LREC*.

- Pamela Forner, Anselmo Peñas, Eneko Agirre, Iñaki Alegria, Corina Forăscu, Nicolas Moreau, Petya Osenova, Prokopis Prokopidis, Paulo Rocha, Bogdan Sacaleanu, et al. 2008. [Overview of the CLEF 2008 multilingual question answering track](#). In *CLEF*.
- Timnit Gebru, J. Morgenstern, Briana Vecchione, Jennifer Wortman Vaughan, H. Wallach, Hal Daumé, and K. Crawford. 2018. [Datasheets for datasets](#). In *FAT/ML*.
- Danilo Giampiccolo, Pamela Forner, Jesús Herrera, Anselmo Peñas, Christelle Ayache, Corina Forascu, Valentin Jijkoun, Petya Osenova, Paulo Rocha, Bogdan Sacaleanu, et al. 2007. [Overview of the CLEF 2007 multilingual question answering track](#). In *CLEF*.
- Momchil Hardalov, Todor Mihaylov, Dimitrina Zlatkova, Yoan Dinkov, Ivan Koychev, and Preslav Nakov. 2020. [EXAMS: A multi-subject high school examinations dataset for cross-lingual and multilingual question answering](#). In *EMNLP*.
- David A. Hull and Gregory Grefenstette. 1996. [Querying across languages: a dictionary-based approach to multilingual information retrieval](#). In *SIGIR*.
- Vladimir Karpukhin, Barlas Oğuz, Sewon Min, Ledell Wu, Sergey Edunov, Danqi Chen, and Wen-tau Yih. 2020. [Dense passage retrieval for open-domain question answering](#). In *EMNLP*.
- Philipp Koehn, Hieu Hoang, Alexandra Birch, Chris Callison-Burch, Marcello Federico, Nicola Bertoldi, Brooke Cowan, Wade Shen, Christine Moran, Richard Zens, Chris Dyer, Ondřej Bojar, Alexandra Constantin, and Evan Herbst. 2007. [Moses: Open source toolkit for statistical machine translation](#). In *ACL System Demonstrations*.
- Taku Kudo. 2006. [MeCab: Yet another part-of-speech and morphological analyzer](#).
- Taku Kudo and John Richardson. 2018. [SentencePiece: A simple and language independent subword tokenizer and detokenizer for neural text processing](#). In *EMNLP System Demonstrations*.
- 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. 2019. [Natural Questions: A benchmark for question answering research](#). *TACL*.
- Kenton Lee, Ming-Wei Chang, and Kristina Toutanova. 2019. [Latent retrieval for weakly supervised open domain question answering](#). In *ACL*.
- Patrick Lewis, Barlas Oğuz, Ruty Rinott, Sebastian Riedel, and Holger Schwenk. 2020. [MLQA: Evaluating cross-lingual extractive question answering](#). In *ACL*.
- Seungyoung Lim, Myungji Kim, and Jooyoul Lee. 2019. [KorQuaAD1.0: Korean QA dataset for machine reading comprehension](#).
- Jiahua Liu, Yankai Lin, Zhiyuan Liu, and Maosong Sun. 2019. [XQA: A cross-lingual open-domain question answering dataset](#). In *ACL*.
- Shayne Longpre, Yi Lu, and Joachim Daiber. 2020. [MKQA: A linguistically diverse benchmark for multilingual open domain question answering](#).
- Bernardo Magnini, Danilo Giampiccolo, Pamela Forner, Christelle Ayache, Valentin Jijkoun, Petya Osenova, Anselmo Penas, Paulo Rocha, Bogdan Sacaleanu, and Richard Sutcliffe. 2006. [Overview of the CLEF 2006 multilingual question answering track](#). In *CLEF*.
- Bernardo Magnini, Simone Romagnoli, Alessandro Vallin, Jesús Herrera, Anselmo Penas, Víctor Peinado, Felisa Verdejo, and Maarten de Rijke. 2003. [The multiple language question answering track at CLEF 2003](#). In *CLEF*.
- Bernardo Magnini, Alessandro Vallin, Christelle Ayache, Gregor Erbach, Anselmo Peñas, Maarten De Rijke, Paulo Rocha, Kiril Simov, and Richard Sutcliffe. 2004. [Overview of the CLEF 2004 multilingual question answering track](#). In *CLEF*.
- Miniwatts Marketing Group. 2011. [Internet world stats: Usage and population statistics](#).
- Hussein Mozannar, Elie Maamary, Karl El Hajal, and Hazem Hajj. 2019. [Neural Arabic question answering](#). In *WANLP*.
- Myle Ott, Sergey Edunov, Alexei Baevski, Angela Fan, Sam Gross, Nathan Ng, David Grangier, and Michael Auli. 2019. [fairseq: A fast, extensible toolkit for sequence modeling](#). In *NAACL System Demonstrations*.
- Kishore Papineni, Salim Roukos, Todd Ward, and Wei-Jing Zhu. 2002. [BLEU: a method for automatic evaluation of machine translation](#). In *ACL*.
- Pranav Rajpurkar, Jian Zhang, Konstantin Lopyrev, and Percy Liang. 2016. [SQuAD: 100,000+ questions for machine comprehension of text](#). In *EMNLP*.
- Stephen Robertson and Hugo Zaragoza. 2009. [The probabilistic relevance framework: BM25 and beyond](#). *Foundations and Trends in Information Retrieval*.
- Uma Roy, Noah Constant, Rami Al-Rfou, Aditya Barua, Aaron Phillips, and Yinfei Yang. 2020. [LARQA: Language-agnostic answer retrieval from a multilingual pool](#). In *EMNLP*.
- Shota Sasaki, Shuo Sun, Shigehiko Schamoni, Kevin Duh, and Kentaro Inui. 2018. [Cross-lingual learning-to-rank with shared representations](#). In *NAACL*.

- Rico Sennrich, Barry Haddow, and Alexandra Birch. 2016. [Neural machine translation of rare words with subword units](#). In *ACL*.
- Jörg Tiedemann and Lars Nygaard. 2004. [The OPUS corpus - parallel and free](#). In *LREC*.
- Jörg Tiedemann and Santhosh Thottingal. 2020. [OPUS-MT — Building open translation services for the World](#). In *EAMT*.
- Alessandro Vallin, Bernardo Magnini, Danilo Giampiccolo, Lili Aunimo, Christelle Ayache, Petya Osenova, Anselmo Peñas, Maarten De Rijke, Bogdan Sacaleanu, Diana Santos, et al. 2005. [Overview of the CLEF 2005 multilingual question answering track](#). In *CLEF*.
- Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. 2017. [Attention is all you need](#). In *NeurIPS*.
- Rui Zhang, Caitlin Westerfield, Sungrok Shim, Garrett Bingham, Alexander R. Fabbri, Neha Verma, William T Hu, and Dragomir R. Radev. 2019. [Improving low-resource cross-lingual document retrieval by reranking with deep bilingual representations](#). In *ACL*.
- Michał Ziemski, Marcin Junczys-Dowmunt, and Bruno Pouliquen. 2016. [The United Nations parallel corpus v1.0](#). In *LREC*.

## Appendix

### A Spirit behind Annotation Interface Design

**Open-retrieval annotation desiderata.** Open-retrieval QA annotation comes with unique challenges. In article-oriented QA such as SQuAD (Rajpurkar et al., 2016), all labels are with regard to a single document and a single human can indeed read the whole document. In open-retrieval QA, answers can be retrieved from millions of documents. Because exhaustively reading so much content is impossible for humans, the notion of “human performance” must be reconsidered in this context. This is why we only evaluate questions having answers in the open-retrieval setting and discard those where no answer was found—it is difficult to prove an answer does not exist in the millions of documents.

**Limits of traditional annotation.** In addition to fundamental problems of information scarcity and asymmetry in multilingual QA, questions can be labeled as unanswerable simply because of annotation errors. Annotation procedures for information-seeking QA data usually have each annotator read a single Wikipedia article retrieved by a search engine and label a correct answer span or label the question as not answered by the article (Kwiatkowski et al., 2019; Clark et al., 2020). In this procedure, the answer coverage is underestimated when the search engine fails to retrieve relevant articles (*retrieval errors*) or the annotator overlooks answer content from the selected articles (*answer annotation errors*, Asai and Choi, 2020). Importantly, these two types of annotation errors present a tradeoff: if we retrieve many articles, retrieval errors will be reduced at the expense of answer annotation errors because annotators have to find answer context among many candidate articles. An annotation procedure that misses too many answers will lead to an artificially small dataset.

### B Additional Details of Dataset Creation

#### B.1 Annotation Interface

In this section, we describe the details of the annotation interface we used for answer annotation in English (§2.1.3). The annotation interface can be seen in Figs. 4 and 5. To maximize the answer coverage for open-retrieval questions, we first rank paragraphs from top articles retrieved by Google



Search. During this paragraph ranking process, we only consider top 5 paragraphs and exclude the articles ranked from top 6 to 10. Increasing the number of the initial articles introduces more noise and confuses our paragraph ranking model, while human annotators sometimes found that those low-ranked articles relevant and retrieved answers from them as discussed in §2.1.3. In the annotation interface, we first present those top 5 paragraphs first (the ones highlighted in light blue in Fig. 4). When annotators do not find answers in the pre-selected top 5 paragraphs, they will explore more paragraphs and articles by expanding originally collapsed articles as in Fig. 5.

Who was the first Prime Minister of Japan?

[Prime Minister of Japan](#) [\[collapse article\]](#) [\[hide full article\]](#)

**Introduction**

The is the head of government and chief executive of Japan. The Prime Minister is appointed by the Emperor of Japan after being designated by the National Diet and must enjoy the confidence of the House of Representatives to remain in office. He is the chairman of the Cabinet and appoints and dismisses the other Ministers of State. The literal translation of the Japanese name for the office is "Minister for the Comprehensive Administration of ("or" the Presidency over) the Cabinet".

**History.**

Before the adoption of the Meiji Constitution, Japan had in practice no written constitution. Originally, a Chinese-inspired legal system known as "ritsuryō" was enacted in the late Asuka period and early Nara period. It described a government based on an elaborate and rational meritocratic bureaucracy, serving, in theory, under the ultimate authority of the Emperor; although in practice, real power was often held elsewhere, such as in the hands of the Fujiwara clan, who intermarried with the Imperial Family in the Heian period, or by the ruling "shōgun". Theoretically, the last "ritsuryō" code, the Yōmei Code enacted in 752, was still in force at the time of the Meiji Restoration.

Under this system, the was the head of the "Daijō-kan" (Department of State), the highest organ of Japan's pre-modern imperial government during the Heian period and until briefly under the Meiji Constitution with the appointment of Sanjō Sanetomi in 1871. The office was replaced in 1885 with the appointment of Itō Hirobumi to the new position of Prime Minister, four years before the enactment of the Meiji Constitution, which mentions neither the Cabinet nor the position of Prime Minister explicitly. It took its current form with the adoption of the Constitution of Japan in 1947.

To date, 62 people have served this position. The current Prime Minister is Shinzo Abe, who re-took office on December 26, 2012. He is the first former Prime Minister to return to office since 1948, and the 5th longest serving Prime Minister to date.

1. Click a paragraph that answers the question.

[No gold paragraph.](#)

2-a. Select a minimal answer span if any.

Itō Hirobumi

2-b. Select a yes/no answer, if that's what the question asks for.

[Yes](#) [No](#)

[Previous](#) [Next](#)

[Ready to Submit HIT](#)

Figure 4: Annotation interface (expanded). The blue highlighted paragraphs are ranked high by the BERT paragraph ranker, and the orange highlighted paragraph is the one clicked by the annotator.

Who was the first Prime Minister of Japan?

[Prime Minister of Japan](#) [\[show paragraph\]](#) [\[show full article\]](#)

[List of Japanese prime ministers by longevity](#) [\[collapse article\]](#)

**Introduction**

This is a list of Japanese prime ministers by longevity. It consists of Prime Ministers and interim Prime Ministers of Japan who have held the office. If a Prime Minister served more than one non-consecutive term, the dates given are for the beginning of their first term, and the end of their last term.

To avoid confusion and maintain consistency, the name of the Prime Ministers are listed in the Western style (given name, family name). Where the person in question is still living, the longevity is calculated up to .

**Overview.**

The median age of a Prime Minister first taking office is 61 years and 2 months. This falls between Tomotsurō Katō and Keizō Obuchi. The youngest Prime Minister was the first, Hirobumi Itō, who took office at the age of 44 years, 67 days. The oldest Prime Minister to first take office was Kantarō Suzuki, who became Prime Minister at the age of 77 years, 79 days.

The oldest living Prime Minister is Yasuhiro Nakasone, born 27 May 1918 (age ). The youngest living former Prime Minister is Yoshihiko Noda, born 20 May 1957 (age ).

The longest-lived Prime Minister was Naruhiko Higashikuni, who died at the age of 102 years, 48 days. If the oldest living Prime Minister, Yasuhiro Nakasone (born 27 May 1918) lives to 13 July 2020, he will tie this record. The shortest-lived Prime Minister was Sanetomi Sanjō, who died at the age of 53 years and 352 days.

[Junichiro Koizumi](#) [\[show article\]](#)

[Naoto Kan](#) [\[show article\]](#)

1. Click a paragraph that answers the question.

[No gold paragraph.](#)

2-a. Select a minimal answer span if any.

Itō Hirobumi

2-b. Select a yes/no answer, if that's what the question asks for.

[Yes](#) [No](#)

[Previous](#) [Next](#)

[Ready to Submit HIT](#)

Figure 5: Annotation interface (collapsed). Annotators can choose to read full articles or collapse articles.

## B.2 Quality Control for Question Translation

We first ask Gengo translators to translate 20 sample questions following our detailed instruction before starting the task, and ask native speakers to assess the quality of translations. We filter out translators who do not provide translation results that meet our standard (e.g., wrong translations of entities, heavy reliance on public machine translation systems). We have found that some of the translators almost copy and paste outputs of existing APIs without fixing errors even when there are crucial errors. After this initial qualification process, we observe that the translation quality is sufficiently high.

## B.3 Quality Control for QA annotation

To control the QA annotation quality, we recruit workers with a high approval rate ( $\geq 96\%$ ) located in English-speaking countries and conducted a rigorous qualification procedure. In our qualification stage, we post small calibration batches and evaluate the workers' performance by expert judgments from authors and agreement with other annotators. To keep the high quality of annotations, we randomly sample qualified workers weekly and manually monitor their annotations by comparing them with gold annotations by authors. We remove qualifications when we detect too many incorrect annotations (e.g., label a paragraph about a different person as a gold paragraph) and remove the annotations done by those disqualified annotators, which are later reannotated by a qualified worker. Over 200 annotators participated in our calibration tasks. About 40 workers are qualified with 24 actively working on the final dataset. Each HIT contains 5 questions with a reward ranging from 1.5 to 2.5 USD. Qualified annotators generally spend 1-2 minutes to answer each question. We give special rewards to annotators who actively search additional paragraphs or articles; the amounts of the rewards are calculated based on the numbers of the HITs they have submitted, resulting in 5-10 USD for each payment.

## B.4 Answer Translation Instructions

During answer translation, we asked annotators to follow the instructions listed below:

- Translators need to use metric units by default, instead of imperial units.
- If the original answers are expressed in an imperial unit, translators are encouraged to

convert them into a metric unit (e.g., Height 5'3" → 身長 160 cm).

- When translating proper nouns, translators are asked to use an official translation if it is available in Wikipedia; otherwise they are encouraged to transliterate them.

We also specify some language-specific instructions to make the translated answers consistent with the ones in the original TYDI QA dataset.

- For Japanese and Korean, translators do not need to spell out the numbers (e.g., 1954 → 千九百五十四) as people usually use Arabic numerals.
- For Bengali, we expect the numbers will be spelled out in Bengali numerals as Bengali speakers rarely use Arabic numerals.
- For Japanese and Korean, translators use appropriate measure words (e.g., 1867년, 57歲) if those measure words are commonly added in those languages.
- For the languages where the date needs to be expressed in some rigid format, translators need to follow the format.

## B.5 Full Data Statistics of Cross-lingual data

Seen in Table 9 are full data statistics of cross-lingual data of XOR-TYDI QA. Among the questions with “Long” answer annotations are some questions without any short answers as in Natural Questions or TYDI QA. We do not include those “Long answer only” examples in our XOR-TYDI QA evaluations.

## C Training details

We describe the details in training our baselines to facilitate easy replication of our results.

### C.1 Machine Translation Models

Table 10 lists hyperparameters for training our transformer machine translation models. We generally follow the hyperparameters for the base-sized transformer (Vaswani et al., 2017). The one exception is English↔Russian where we used pretrained transformer large models.<sup>11</sup> For each language direction, all data are encoded into subwords by Moses tokenization (Koehn et al., 2007, for Arabic, Finnish, and Russian) and BPE (Sennrich et al., 2016) or SentencePiece (Kudo and Richardson, 2018, for

Bengali, Japanese, Korean, and Telugu). We train an autoregressive transformer (Vaswani et al., 2017) with the fairseq library (Ott et al., 2019).

### C.2 Retrieval Models

**Training DPR and Path Retriever.** To train an English DPR and Path Retriever, we first initialize the parameters of the models with the ones trained on Natural Questions Open data, which is available on their repository. During finetuning on XOR-TYDI QA, we use the human translated questions with the annotated gold paragraph data.

**Choice of negative and positive context.** Selection of positive and negative examples is crucial to train competitive neural retriever models (Karpukhin et al., 2020). We follow the hyperparameters used in the original papers (Karpukhin et al., 2020; Asai et al., 2020). To construct effective negative and positive context, we follow the approaches introduced by the authors of those works.

To train DPR, we use the original gold paragraphs (long answers) annotated by MTurkers as positive passages. Following the experimental settings of DPR on Natural Questions, we first split gold paragraphs into 100-token units, and consider the units with the original short answer annotations as positive context. For negative context, we first randomly sample one negative paragraph per question from the top 5 paragraphs pre-selected by our paragraph reranking model in §2.1.3, split the negative paragraph into 100-token units, and then randomly pick one to use it as a negative context. We also reuse the in-batch negative paragraphs as discussed in Karpukhin et al. (2020).

Regarding the training of Path Retriever, we randomly sample top 50 paragraphs from the top 10 articles retrieved for annotations and use them as negative paragraphs. We also use the annotated long answers as positive paragraphs.

### Implementation details of BM25 Retrievers.

To implement BM25-based retrievers for the 7 languages, we use ElasticSearch’s Python client (Python Elasticsearch Client).<sup>12</sup> We apply the default tokenizers and analyzers for Arabic, Bengali, Finnish and Russian. Japanese and Korean are not supported by the default ElasticSearch language

<sup>11</sup><https://github.com/pytorch/fairseq/blob/master/examples/translation/README.md>.

<sup>12</sup><https://elasticsearch-py.readthedocs.io/en/master/>.

$L_i$	Train (1 way)			Dev (2 way)			Test (2 way)		
	Total	Long (%)	Short (%)	Total	Long (%)	Short (%)	total	Long (%)	Short (%)
Arabic	4,500	2,862 (63)	2,574 (57)	500	357 (71)	350 (70)	235	144 (61)	137 (58)
Bengali	4,500	2,822 (63)	2,582 (57)	500	330 (66)	312 (62)	185	131 (70)	128 (69)
Finnish	4,500	2,454 (55)	2,088 (46)	500	372 (74)	360 (72)	800	556 (69)	530 (66)
Japanese	4,500	2,557 (57)	2,288 (51)	500	320 (64)	296 (60)	779	477 (61)	449 (58)
Korean	4,500	2,674 (59)	2,469 (55)	500	314 (63)	299 (60)	1,177	684 (58)	646 (55)
Russian	4,500	2,178 (48)	1,941 (43)	500	270 (54)	255 (51)	470	252 (53)	235 (50)
Telugu	4,500	1,515 (33)	1,308 (29)	500	258 (52)	238 (47)	1,752	394 (22)	374 (21)

Table 9: Dataset statistics of the resulting XOR QA corpus (cross-lingual data only). “Long” denotes the questions with paragraph answer annotations, and “Short” denotes the questions with short answer annotations. During evaluation, we disregard the questions without short answer annotations.

Hyperparameter	Value
label smoothing	0.1
# max tokens	4096
dropout rate	0.3
encoder embedding dim	512
encoder ffn dim	2048
# encoder attn heads	8
decoder embedding dim	512
decoder ffn dim	2048
# decoder attn heads	8
max source positions	10000
max target positions	10000
Adam lr rate	$5 \times 10^{-4}$
Adam $\beta_1$	0.9
Adam $\beta_2$	0.98
lr-scheduler	inverse square
warm-up lr	$1 \times 10^{-7}$
# warmup updates	4000
# max updates	300K
length penalty	1.0

Table 10: Hyperparameters for our transformer machine translation models.

analyzers, so we use Kuromoji<sup>13</sup> and Nori plugins<sup>14</sup> for Japanese and Korean respectively. Note that we do not implement a BM25-based retriever for Telugu, since it is not supported by the default language analyzer and we could not find an official plugin for Telugu.

### C.3 Machine Reading Models

We use the official hyperparameters for machine reading components of DPR and Path Retriever. Table 11 shows the list of the hyperparameters used to train a multilingual machine reading model for the monolingual pipeline in XOR-FULL. We lowercased input paragraphs and questions.

<sup>13</sup><https://www.elastic.co/guide/en/elasticsearch/plugins/7.9/analysis-kuromoji.html>.

<sup>14</sup><https://www.elastic.co/guide/en/elasticsearch/plugins/7.9/analysis-nori.html>.

Hyperparameter	Value
max sequence length	384
document stride	128
max query length	64
Adam lr rate	$5 \times 10^{-4}$
Adam $\epsilon$	$1 \times 10^{-8}$
max gradient norm	1.0
# train epochs	3.0
seed	42

Table 11: Hyperparameters for our machine reading model in the monolingual pipeline.

**Choice of negative and positive examples.** For the Path Retriever and BM25 baselines’ reader, we sample three negative paragraphs per annotated question-gold paragraph pair and train a model that jointly predicts an answer span and relevance score of each paragraph to the question, following Asai et al. (2020). In DPR, the training examples are retrieved by the trained retriever, and we train the reader with 24 negative paragraphs by distant supervision (Karpukhin et al., 2020). We use human translated English questions to train English reader models, and use the original questions in  $L_i$  to train a multilingual reader model.

## D Additional Results and Analysis

### D.1 Additional Experimental Results

**XOR-RETRIEVE.** We present the R@2kt scores of the retrieval baselines in Table 12. As shown in Table 5, given human translations, DPR generally outperforms other two retrieval baselines. We also present R@2kt and R@5kt of our DPR models on our development set in Table 13, and we observe a similar performance trend to the test set: models with queries translated by GMT outperform other models in all of the XOR-TYDI QA languages. Comparing the two baselines that do not use external black-box APIs, we see that

	Human			GMT		Our MT		<i>Multi.</i>
	DPR	PATH	BM	DPR	PATH	DPR	PATH	
Ar	<b>65.8</b>	65.0	41.6	61.7	59.1	48.3	45.0	41.2
Bn	72.8	<b>78.1</b>	57.7	72.0	58.2	54.4	60.9	43.9
Fi	66.5	<b>68.0</b>	43.7	60.6	60.3	56.7	56.6	50.3
Ja	<b>62.0</b>	59.0	38.8	52.1	50.0	41.8	36.7	29.1
Ko	<b>65.0</b>	60.0	43.8	57.9	50.3	39.4	33.8	34.5
Ru	57.5	<b>59.9</b>	35.2	51.2	54.1	39.6	34.7	35.3
Te	<b>66.3</b>	59.6	44.6	59.4	58.0	18.7	15.7	37.2
Av.	<b>65.1</b>	64.3	43.5	59.3	58.2	42.7	40.5	38.8

Table 12: R@2kt (§3.1) on the test data in the XOR-RETRIEVE setting. PATH and BM denote Path Retriever and BM25 respectively. The rightmost column is a multilingual approach that bypasses the query translation step (§3.1).

	GMT		Our MT		<i>Multi.</i>	
	R@2kt	R@5kt	R@2kt	R@5kt	R@2kt	R@5kt
Ar	62.5	69.6	43.4	52.4	38.8	48.9
Bn	74.7	82.2	53.9	62.8	48.4	60.2
Fi	57.3	62.4	55.1	61.8	52.5	59.2
Ja	55.6	64.7	40.2	48.1	26.6	34.9
Ko	60.0	68.8	50.5	58.6	44.2	49.8
Ru	52.7	60.8	30.8	37.8	33.3	43.0
Te	72.3	79.0	20.2	32.4	39.9	55.5
Av.	62.2	69.6	42.0	50.6	40.5	50.2

Table 13: R@5kt (§3.1) of DPR models (translate DPR and multilingual DPR) on the development data in the XOR-RETRIEVE setting.

the translation approach (Our MT) outperforms the multilingual one (*Multi.*) in Arabic, Bengali, Finnish Japanese, and Korean, while it performs poorly in Telugu. These results are consistent with the ones on the test data in Table. 5.

**XOR-ENGLISHSPAN.** Table 14 shows the F1 and EM scores of our DPR models on the development data in the XOR-ENGLISHSPAN setting. Similar to the results on XOR-RETRIEVE, GMT significantly outperforms our MT and our multilingual model. Probably due to the error propagation, the Telugu performance of our MT baseline is low, indicating the importance of developing a multilingual baseline that could perform well on languages with little parallel data for translation training.

**XOR-FULL.** We present F1, BLEU and EM scores for XOR-FULL in Tables 15, 16 and 17. We also present F1 scores and average F1, BLEU and EM scores on the development set in Table 18.

	GMT		Our MT		<i>Multi.</i>	
	F1	EM	F1	EM	F1	EM
Ar	35.4	27.7	20.9	14.9	17.2	12.3
Bn	42.1	35.3	25.2	20.5	21.8	17.3
Fi	31.8	23.1	31.9	23.3	27.6	20.8
Ja	27.2	20.9	19.6	15.5	15.5	12.8
Ko	32.5	22.7	25.3	18.1	18.5	14.4
Ru	34.7	28.2	16.1	11.4	21.3	17.3
Te	35.0	27.4	3.6	1.7	17.7	13.1
Av.	35.0	27.4	20.4	15.1	19.9	15.4

Table 14: F1 and EM scores of our DPR models (translate DPR and multilingual DPR) on the development data in the XOR-ENGLISHSPAN setting.

## D.2 Additional Analysis

**Single language Wikipedia ablations in XOR-FULL.** In XOR-FULL, a system is expected to answer a question in the target language by consulting multilingual Wikipedia corpora, but which language answer content exists in is not known *a priori* (§3.3). To understand the benefit of retrieving evidence from a multilingual document pool, we run single language Wikipedia ablations. In this study, we conduct ablations in which systems only use either English Wikipedia ( $W_{eng}$ ) or the target language’s Wikipedia ( $W_i$ ). We run the monolingual baselines for  $W_i$  only and the cross-lingual baselines for  $W_{eng}$  only. For the cross-lingual baseline, all predicted answers will be translated back to the target languages.

The bottom section of Table 15 shows the full results of single language Wikipedia ablations on XOR-FULL. In a majority of the languages, we observed performance drops from the full models that use both  $W_{eng}$  and  $W_i$  (e.g., 20.1 vs. 14.3 F1 in Korean). In Japanese and Finnish, our English Wikipedia only baselines outperform the full models. Currently, the answer aggregation process prioritizes answers predicted by monolingual models, but the monolingual models perform poorly in those two languages. Future work can address the challenges of improving evidence and answer aggregation from multilingual document collections.

**Per-difficulty retrieval performance.** We split our data by annotation difficulty i.e., whether or not a gold paragraph is selected by the BERT retriever used during annotation in our collaborative annotation framework (§2.1.3). Table 19 presents retrieval performance broken down by difficulty. We observed a large performance gap between the easy and hard subsets (65.3 for easy vs. 59.9 for



Wiki Corpus	Translation		Retrieval		Target Language $L_i$							
	Query	Answer	$L_i$	Eng.	Ar	Bn	Fi	Ja	Ko	Ru	Te	Avg.
$W_{i,eng}$	GMT	GMT	GS	DPR	<b>31.5</b>	19.0	18.3	8.8	<b>20.1</b>	<b>19.8</b>	<b>13.6</b>	18.7
	Our MT	Our MT	GS	DPR	29.6	6.6	15.5	7.6	16.4	18.7	1.7	13.7
	Our MT	Our MT	BM25	DPR	12.1	<b>22.0</b>	9.3	5.4	9.7	7.4	0.8	9.5
	–	GMT	GS	mDPR	30.5	5.2	16.9	8.2	17.6	<b>19.8</b>	6.0	15.7
$W_{eng}$	GMT	GMT	–	DPR	23.9	18.5	<b>22.9</b>	<b>24.1</b>	17.5	16.8	13.2	<b>19.5</b>
	Our MT	Our MT	–	DPR	7.6	5.9	16.2	9.0	5.3	5.5	0.8	7.2
	–	GMT	–	mDPR	12.4	9.7	19.1	14.0	8.2	10.9	5.4	11.3
$W_i$	–	–	GS	–	29.0	0.9	9.5	6.2	14.3	18.5	0.9	11.3
	–	–	BM25	–	12.0	<b>22.0</b>	9.3	5.3	9.7	7.4	–	–

Table 15: Performance on XOR-FULL task (F1 scores on the test data). “GS” denotes Google Search retrieval. The bottom section shows results from single Wikipedia baselines. Elasticsearch for BM25 does not support Telugu. “mDPR” denotes a DPR model where query and context encoders are initialized with multilingual BERT.

Wiki Corpus	Translation		Retrieval		Target Language $L_i$							
	Query	Answer	$L_i$	Eng.	Ar	Bn	Fi	Ja	Ko	Ru	Te	Avg.
$W_{i,eng}$	GMT	GMT	GS	DPR	<b>22.1</b>	10.9	13.3	3.0	<b>20.1</b>	<b>11.4</b>	<b>9.1</b>	<b>12.1</b>
	Our MT	Our MT	GS	DPR	20.9	2.2	10.9	2.3	12.6	10.5	1.4	8.7
	Our MT	Our MT	BM25	DPR	7.7	<b>15.4</b>	6.4	1.3	6.7	3.9	0.6	6.0
	–	GMT	GS	mDPR	21.4	5.2	12.1	2.7	13.3	11.3	3.9	10.0
$W_{eng}$	GMT	GMT	–	DPR	12.3	10.1	<b>16.6</b>	<b>14.1</b>	11.5	10.4	8.5	12.0
	Our MT	Our MT	–	DPR	2.5	1.5	10.3	3.3	2.9	2.5	0.5	3.4
	–	GMT	–	mDPR	6.7	4.5	13.5	8.1	8.2	6.5	3.1	6.8
$W_i$	–	–	GS	–	20.6	0.7	7.1	1.5	11.5	10.4	0.8	7.5
	–	–	BM25	–	7.7	15.3	6.4	1.3	6.7	3.9	–	–

Table 16: Performance on XOR-FULL (EM scores on the test data). “GS” denotes Google Search retrieval. The bottom section shows results from single Wikipedia baselines. Elasticsearch for BM25 does not support Telugu. “mDPR” denotes a DPR model where query and context encoders are initialized with multilingual BERT.

Wiki Corpus	Translation		Retrieval		Target Language $L_i$							
	Query	Answer	$L_i$	Eng.	Ar	Bn	Fi	Ja	Ko	Ru	Te	Avg.
$W_{i,eng}$	GMT	GMT	GS	DPR	<b>29.7</b>	22.1	18.8	2.2	<b>13.3</b>	<b>18.0</b>	<b>13.5</b>	<b>16.8</b>
	Our MT	Our MT	GS	DPR	27.8	7.4	10.9	2.0	12.6	17.0	1.1	8.9
	Our MT	Our MT	BM25	DPR	12.8	<b>22.9</b>	6.4	1.2	7.0	7.3	0.3	12.0
	–	GMT	GS	mDPR	27.8	7.0	13.9	1.8	11.3	17.0	5.3	13.9
$W_{eng}$	GMT	GMT	–	DPR	24.5	21.4	<b>20.6</b>	<b>6.1</b>	10.0	14.2	13.3	15.7
	Our MT	Our MT	–	DPR	8.6	6.7	16.7	2.8	3.9	5.0	0.3	6.3
	–	GMT	–	mDPR	12.2	10.2	16.7	2.4	4.7	8.2	8.3	9.0
$W_i$	–	–	GS	–	27.3	0.7	10.4	1.6	10.4	16.6	0.8	9.7
	–	–	BM25	–	12.8	<b>22.9</b>	10.6	1.2	7.0	7.3	–	–

Table 17: Performance on XOR-FULL (BLEU scores on the test data). “GS” denotes Google Search retrieval. The bottom section shows results from single Wikipedia baselines. Elasticsearch for BM25 does not support Telugu. “mDPR” denotes a DPR model where query and context encoders are initialized with multilingual BERT.

Translation		Retrieval		Target Language $L_i$								Macro Average		
Query	Answer	$L_i$	Eng.	Ar	Bn	Fi	Ja	Ko	Ru	Te		F1	EM	BLEU
GMT	GMT	GS	DPR	<b>18.0</b>	<b>29.1</b>	13.8	<b>5.7</b>	<b>15.2</b>	14.9	<b>15.6</b>		<b>16.0</b>	<b>9.9</b>	<b>14.9</b>
Our MT	Our MT	GS	DPR	17.7	4.5	13.0	<b>5.7</b>	15.0	14.9	8.8		11.4	6.3	10.3
Our MT	Our MT	BM25	DPR	9.2	15.8	<b>14.4</b>	4.8	7.9	5.2	0.5		8.3	4.6	7.5
–	GMT	GS	mDPR	17.8	15.3	12.6	5.6	<b>15.2</b>	<b>15.0</b>	10.1		13.1	7.7	12.2

Table 18: Performance on XOR-FULL (dev data F1 scores and average F1, EM and BLEU scores). “GS” denotes Google Search retrieval, and “mDPR” denotes a DPR model where query and context encoders are initialized with multilingual BERT.

Query Translator	Easy		Hard	
	R@2kt	R@5kt	R@2kt	R@5kt
Human	65.3	72.5	59.9	68.9
GMT	61.1	67.7	54.3	63.4
Our MT	41.9	49.9	37.7	44.5
<i>Multilingual</i>	34.3	44.3	36.1	40.9

Table 19: Macro-averaged retrieval recall on the *easy* and *hard* subsets of the development set. All configurations use DPR for retrieval. The *Multilingual* model avoids query translation.

hard subsets in R@2kt with human translation and DPR), suggesting that the questions from the hard subset are clearly more challenging than the ones from the easy subset.

### D.3 Qualitative Analysis on Translation Errors

One primary challenge in question translation is precisely translating key words (e.g., entities, year); our MT correctly translates a Japanese question, アーモンドアイはいつ生まれた？ (*When was Almond Eye born*; Almond Eye is a Japanese popular race horse)<sup>15</sup> while Helsinki (Tiedemann and Thottingal, 2020) translates it to “When was almond born?” This resulted in retrieval errors, and Wikipedia articles related to almonds were selected. Intrinsic metrics such as BLEU would not consider the importance of these translation mistakes.

<sup>15</sup>[https://en.wikipedia.org/wiki/Almond\\_Eye](https://en.wikipedia.org/wiki/Almond_Eye).