# IDK-MRC: Unanswerable Questions for Indonesian Machine Reading Comprehension

**Rifki Afina Putri**
School of Computing
KAIST, South Korea
rifkiaputri@kaist.ac.kr

**Alice Oh**
School of Computing
KAIST, South Korea
alice.oh@kaist.edu

## Abstract

Machine Reading Comprehension (MRC) has become one of the essential tasks in Natural Language Understanding (NLU) as it is often included in several NLU benchmarks (Liang et al., 2020; Wilie et al., 2020). However, most MRC datasets only have answerable question type, overlooking the importance of unanswerable questions. MRC models trained only on answerable questions will select the span that is most likely to be the answer, even when the answer does not actually exist in the given passage (Rajpurkar et al., 2018). This problem especially remains in medium- to low-resource languages like Indonesian. Existing Indonesian MRC datasets (Purwarianti et al., 2007; Clark et al., 2020) are still inadequate because of the small size and limited question types, i.e., they only cover answerable questions. To fill this gap, we build a new Indonesian MRC dataset called I(n)don'tKnow-MRC (IDK-MRC) by combining the automatic and manual unanswerable question generation to minimize the cost of manual dataset construction while maintaining the dataset quality. Combined with the existing answerable questions, IDK-MRC consists of more than 10K questions in total. Our analysis shows that our dataset significantly improves the performance of Indonesian MRC models, showing a large improvement for unanswerable questions[1].

## 1 Introduction

Machine Reading Comprehension (MRC) is a task where a machine is asked to read a given passage and answer a question based on the passage. Several English MRC datasets have been widely used, including SQuAD (Rajpurkar et al., 2016) and NewsQA (Trischler et al., 2017). However, MRC models that do well on those datasets are not guaranteed to be robust. Rajpurkar et al. (2018) highlights the problem of the SQuAD dataset that only focuses on answerable questions, making the model trained on this dataset tends to select the span without carefully checking whether the passage actually has the answer. SQuAD 2.0 is then built by manually adding new unanswerable questions to the existing SQuAD dataset (Rajpurkar et al., 2018).

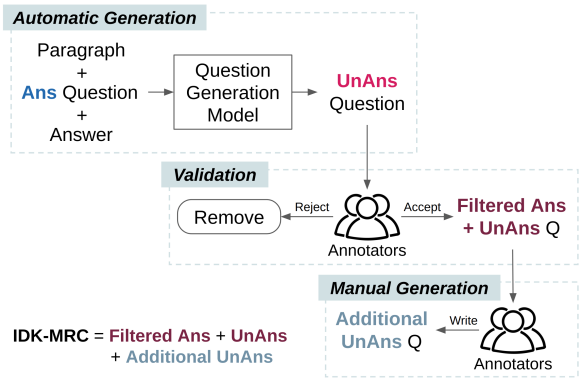


Figure 1: Our dataset collection pipeline.

While SQuAD 2.0 is widely used for evaluation of English models, similar datasets for other languages are still limited, hindering the progress of MRC task for these languages. Indonesian has around 198 million speakers[2], but despite its popularity, there exists an insufficient amount of Indonesian MRC datasets. For instance, FacQA dataset (Purwarianti et al., 2007) has only around 3K samples, and TyDiQA-GoldP dataset (Clark et al., 2020) has around 5K samples. Furthermore, both datasets only have answerable question type, ignoring the importance of incorporating unanswerable questions. Therefore, building an Indonesian MRC dataset that covers unanswerable questions is necessary.

One alternative to construct a new dataset is man-

[1]The code and dataset of IDK-MRC are available at https://github.com/rifkiaputri/IDK-MRC

[2]https://www.babbel.com/en/magazine/how-many-people-speak-indonesian-where-is-it-spoken (Accessed Jan 2022)



*Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing*, pages 6918 - 6933
December 7-11, 2022 

| Type | Description | | Example |
|---|---|---|---|
| Negation | Negation word inserted or removed | Context | Kambing memiliki lemak dalam kandungan susunya. *(Goats have fat in their milk.)* |
| | | Ans Q | Apakah kandungan yang ada dalam susu kambing? *(What are the ingredients in goat's milk?)* |
| | | UnAns Q | Apakah kandungan yang **tidak** ada dalam susu kambing? *(What are the ingredients that **do not** exist in goat's milk?)* |
| Antonym | Antonym used | Context | Aristokrasi adalah sebuah kelas sosial yang **tertinggi** di masyarakat. *(Aristocracy is the **highest** social class in society.)* |
| | | Ans Q | Apakah nama kelas sosial **tertinggi**? *(What is the name of the **highest** social class?)* |
| | | UnAns Q | Apa nama kelas sosial **terendah**? *(What is the name of the **lowest** social class?)* |
| Entity Swap | Entity, date, number, or term replaced with other entity, date, number, or term | Context | Salah satu kandidat standar untuk 4G yang dikomersilkan di dunia yaitu standar Long Term Evolution (LTE) (Swedia sejak 2009). *(One of the standards for 4G commercialized in the world is the Long Term Evolution (LTE) standard (Sweden since 2009).)* |
| | | Ans Q | Di manakah **LTE** pertama kali diciptakan? *(Where was **LTE** first invented?)* |
| | | UnAns Q | Di manakah **3G** pertama diciptakan? *(Where was **3G** first invented?)* |
| Question Tag Swap | Question tag replaced with other question tag | Context | Suaka margasatwa Muara Angke adalah sebuah kawasan konservasi di wilayah hutan bakau di pesisir utara Jakarta. *(Muara Angke Wildlife Sanctuary is a conservation area in the mangrove forest area on the north coast of Jakarta.)* |
| | | Ans Q | **Di mana** Suaka margasatwa Muara Angke dibangun? *(**Where** was Muara Angke Wildlife Sanctuary built?)* |
| | | UnAns Q | **Kapan** Suaka margasatwa Muara Angke dibangun? *(**When** was Muara Angke Wildlife Sanctuary built?)* |
| Specific Condition | Asks for specific condition that is not satisfied by the information in the paragraph | Context | Bon Jovi terdiri dari Vokalis Jon Bon Jovi, Keyboardist David Bryan, Drummer Tico Torres, Gitaris Phil X, dan Bassist Hugh McDonald. *(Bon Jovi consists of Vocalist Jon Bon Jovi, Keyboardist David Bryan, Drummer Tico Torres, Guitarist Phil X, and Bassist Hugh McDonald.)* |
| | | Ans Q | Siapa personil Bon Jovi? *(Who are the members of Bon Jovi?)* |
| | | UnAns Q | Siapa personil Bon Jovi **yang paling jarang dikenal**? *(Who is **the least known member** of Bon Jovi?)* |
| Other | Other cases where the paragraph does not imply any answer | Context | Patrick Star adalah seekor bintang laut yang bersahabat dengan Spongebob. *(Patrick Star is a starfish whose best friend is Spongebob.)* |
| | | Ans Q | Siapakah **teman baik** karakter SpongeBob SquarePants? *(Who is SpongeBob SquarePants' **best friend**?)* |
| | | UnAns Q | Siapa **teman kecil** karakter Spongebob SquarePants? *(Who is Spongebob SquarePants' **childhood friend**?)* |

Table 1: Unanswerable question types that are covered in IDK-MRC.

ually adding the unanswerable questions. This, however, is expensive and time-consuming. Several Question Generation (QG) approaches have been proposed to mitigate this, but most are focused on generating answerable questions (Heilman and Smith, 2010; Du et al., 2017; Du and Cardie, 2018; Klein and Nabi, 2019; Alberti et al., 2019; Kumar et al., 2019; Puri et al., 2020; Shakeri et al., 2020), with only one generating unanswerable questions (Zhu et al., 2019). These models can quickly generate many questions, but the resulting questions are usually less fluent and less relevant to the passage than human-written questions.

This work intends to combine the best of both worlds by incorporating humans into the automatic dataset generation pipeline. Figure 1 shows our pipeline, which consists of three phases: automatic generation, validation, and manual generation. To sum up, our contributions are as follows:

- We construct a new Indonesian MRC dataset called I(n)don'tKnow-MRC (IDK-MRC), consisting of over 5K unanswerable questions with diverse question types, as shown in Table 1. To the best of our knowledge, IDK-MRC is the first Indonesian MRC dataset covering answerable and unanswerable questions.

- We propose a simple dataset collection pipeline consisting of automatic and manual dataset generation. We show that relying *only*

on automatic generation results in highly imbalanced question type distribution; our manual generation method covers this limitation.

- We validate our dataset on the downstream task and show that it effectively improves the MRC models' performance, especially in predicting the answer to the unanswerable questions.

## 2 Related Work

**Existing Indonesian MRC Dataset** While many MRC datasets are available in English (Rajpurkar et al., 2016; Trischler et al., 2017; Rajpurkar et al., 2018), the number of publicly available Indonesian MRC datasets is very limited. A shortcut to obtain Indonesian MRC data is by machine translating English MRC dataset (Muis and Purwarianti, 2020), but it will result in translation artifacts. We may avoid this by recruiting human annotators to translate them manually; still, it leads to *translationese*, where the translated text appears awkward or unnatural (Clark et al., 2020). FacQA (Purwarianti et al., 2007) is part of the IndoNLU benchmark (Wilie et al., 2020) built from a news article. It has around 3K answerable questions, with limited categories of questions: date, location, name, organization, person, and quantitative. Another dataset called TyDiQA-GoldP (Clark et al., 2020), a multilingual QA dataset constructed from Wikipedia, has about 5K Indonesian samples. It also only focuses on answerable questions. To this date, there are no publicly available Indonesian MRC datasets that include unanswerable question type.

**Human-Model Dataset Construction** Combining human and model in dataset construction is mainly applied to adversarial data, such as AdversarialQA (Bartolo et al., 2020) and AdversarialNLI (Nie et al., 2020). In this dynamic adversarial data collection, human annotators are asked to construct adversarial questions to fool the model. Such human-model annotation pipeline has not been tried for unanswerable questions. Wang et al. (2021) analyzed the cost of different dataset labeling strategies, including the combination of GPT-3 (Brown et al., 2020) and human labeling. Although they included the MRC task in their experiment, they only focused on SQuAD 1.1 (Rajpurkar et al., 2016), which only has answerable questions. The effectiveness of the human-model labeling in the context of unanswerable questions remains unclear.

**Unanswerable Question Generation** Various approaches have been proposed for generating answerable questions in English (Heilman and Smith, 2010; Du et al., 2017; Du and Cardie, 2018; Lewis et al., 2019; Klein and Nabi, 2019; Alberti et al., 2019; Puri et al., 2020; Shakeri et al., 2020), Indonesian (Muis and Purwarianti, 2020), and cross- or multi-lingual (Kumar et al., 2019; Chi et al., 2020; Shakeri et al., 2021; Riabi et al., 2021). The question generation technique also applied to generate adversarial questions (Bartolo et al., 2021). However, for unanswerable question generation, the number of works are limited. Zhu et al. (2019) proposed Pair-to-Sequence (Pair2Seq) model that uses separate encoders for the paragraph and answerable question. They utilized English word embedding (i.e., GloVe (Pennington et al., 2014)) and character embedding as the feature and bi-LSTM (Hochreiter and Schmidhuber, 1997) as the encoder. Although their model performed better compared to the rule-based and TF-IDF baselines, it still relied on a traditional word embedding representation as the feature. Differing from their approach, we utilized mT5 model (Xue et al., 2021) that covers contextual representation of 101 languages, including Indonesian. Our experiment (§5.1) confirms that our model outperforms Pair2Seq model, demonstrating the advantage of our approach.

## 3 Dataset Collection Pipeline

We build IDK-MRC dataset by combining model-generated unanswerable questions with human-written questions. As shown in Figure 1, our dataset collection has three stages: automatic generation, validation, and manual generation.

### 3.1 Automatic Generation

In this stage, we automatically construct unanswerable questions using a Question Generation (QG) model. We use translated SQuAD 2.0 (Rajpurkar et al., 2018) as the training data of the QG model. In the inference step, we use the answerable questions from TyDiQA-GoldP (Clark et al., 2020) as a starting point to add more unanswerable questions for our dataset. Our QG model architecture is illustrated in Figure 2.

**Candidate Generation** We utilize mT5 model (Xue et al., 2021) to generate the unanswerable question candidates. We apply `generate unans` prefix, followed by *context*, *answerable question*, and *answer* as the input. Then, using top-p and

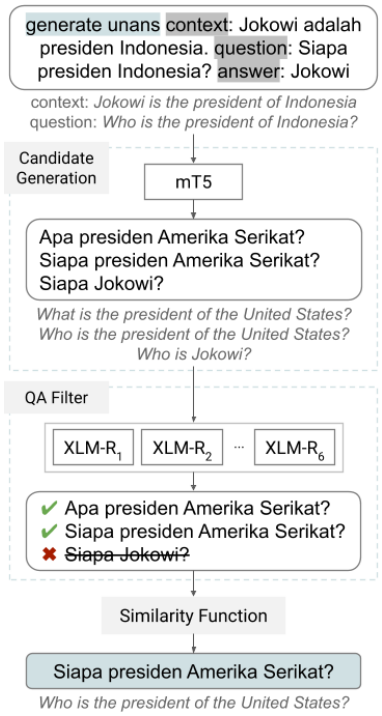


Figure 2: Our proposed question generation model.

top-k sampling as the decoding method, the model produces several output candidates.

**QA Filter** Since not all output candidates are valid unanswerable questions, we filter out invalid questions using an ensemble of six[3] Question Answering (QA) models. We fine-tuned XLM-R (Conneau et al., 2020) on translated SQuAD 2.0 dataset using different random seeds and used them as the QA models. Based on the prediction of these models, we keep the question if four or more models give an empty answer (i.e., unanswerable) or if four or more models return non-empty answers and all these answers are different.

**Similarity Function** Finally, we apply a similarity function to all remaining output candidates to make sure that the final output is relevant to its corresponding paragraph and answerable question. We calculate similarity between the original answerable question and the remaining question candidates using BLEU score to get the unanswerable question with highest n-gram overlap. We pick the candidate with the highest score as the final output.

[3]6 was chosen based on related work in Adversarial QA (Bartolo et al., 2021).

### 3.2 Validation

After obtaining the automatically generated unanswerable questions, we validate them to ensure that they do not have noise or error. We recruit four Indonesian native speakers with 2+ years of experience in Indonesian NLP dataset annotation. Each annotator is asked to give a score to the generated questions with three criteria, adopted from Zhu et al. (2019) and re-defined as follows:

- **Unanswerability**: whether the answer can be found in the given paragraph. The score is 1 if the answer cannot be found, 0 otherwise.
- **Relevancy**: whether the question is relevant to the paragraph and the answerable question. 3 if the question is relevant to both, 2 if it is only relevant to the paragraph or the answerable question, and 1 if it is not relevant to either.
- **Fluency**: whether the question is fluent. 3 if the collective quality of all words in the question is fluent and coherent; 2 if the question is semi-coherent, has a minor typo, or grammatical errors; and 1 if the question is incoherent or incomprehensible.

Each question is validated by one annotator, with each annotator validating the same number of questions. Then, we apply cross-checking method to minimize human errors and to ensure consistency of the criteria across the annotators. Suppose that we have four annotators $(a_1, ..., a_4)$, who have evaluated some set of samples $(s_1, ..., s_4)$. Each sample $s_i$ consists of a set of *paragraph*, *answerable*, *unanswerable question*, along with the *unanswerability*, *relevancy*, and *fluency* scores of the unanswerable question. In the cross-checking phase, $a_1$ is assigned to check the scores of $s_2$, $a_2$ is assigned to check the scores of $s_1$, and so on. The disagreement[4] is resolved by discussion among the annotators to ensure each annotator has the same level of task understanding and thus resulting in high quality and consistent annotation.

Finally, we keep the questions with perfect unanswerability, relevancy, and fluency scores (i.e., questions with scores of 1, 3, 3). We also keep the questions with scores of (1, 3, 2) and ask the annotators to make minor corrections to those questions. We regard the rest of the automatically generated

[4]Overall, the disagreement percentage is roughly around 10–20%, with ∼84% of the disagreement are categorized as narrow disagreement (1 vs 2 or 2 vs 3).

questions as noise and discard them[5]. From 6,196 generated questions, 3,190 questions are kept in the dataset, with 2,840 questions have a perfect score, and 350 questions have 1, 3, and 2 scores for unanswerability, relevancy, and fluency.

### 3.3 Manual Generation

In the final stage, we ask human annotators to add more unanswerable questions, especially for the question types that QG model struggles to generate. There are six question types, as listed in Table 1, and it is important to have a sufficient number of questions for each type. The model generates entity swap questions well (see Figure 3), so we request the annotators to write the remaining question types, i.e., *negation*, *antonym*, *question tag swap*, *specific condition*, and *other*. The annotators may also add a new answerable question to be paired with the *negation* question, specifically for the case when the negation word in the answerable question is removed. The annotators are the same as those from the validation stage, and they were assigned to write 500 unanswerable questions each. We also apply the same cross-checking method as the validation stage. In total, we have 2,000 human-written unanswerable questions.

## 4 The Resulting Data

As shown in Table 2, we have different dataset variations from each dataset collection stage:

- **TyDiQA**: original answerable questions from TyDiQA-GoldP (Clark et al., 2020)[6]. Since the test set is not publicly shared, we made a new split from the existing train and dev data.
- **Model Gen**: unanswerable questions output from the automatic generation stage (§3.1) combines with the answerable questions from TyDiQA. We remove questions that are same as the answerable questions[7].

[5]We remove irrelevant questions (questions with relevancy score of 1 or 2) because they are often too far from the context, making the task less challenging. For example, the context describes Ecology definition, the Ans Q is *"what is the definition of **ecology**?"*, and the UnAns Q is *"what is the definition of **neo**?"*. We remove this kind of question. Note that we still regard the entity as relevant (and keep the question) if it belongs to the same category, i.e., person name, country, etc.

[6]All questions in TyDiQA-GoldP are extractive.

[7]The questions that are the same as the answerable questions are still exist due to the problem in the decoding process in the automatic generation process. We attempt to eliminate this by adding QA Filter model; however, the QA model is not 100% perfect. Therefore, this kind of questions can still remain after the automatic filtering.

| Split | Type | Ans | UnAns | Total |
|---|---|---|---|---|
| Train | TyDiQA | 5,369 | 0 | 5,369 |
| | Model Gen | 5,369 | 5,353 | 10,722 |
| | Human Filt | 4,865 | 2,730 | 7,595 |
| | IDK-MRC | 5,042 | 4,290 | 9,332 |
| Dev | TyDiQA | 402 | 0 | 402 |
| | Model Gen | 402 | 401 | 803 |
| | Human Filt | 364 | 211 | 575 |
| | IDK-MRC | 382 | 382 | 764 |
| Test | TyDiQA | 423 | 0 | 423 |
| | Model Gen | 423 | 423 | 846 |
| | Human Filt | 405 | 249 | 654 |
| | IDK-MRC | 422 | 422 | 844 |

Table 2: Statistics of the datasets.

- **Human Filt**: Model Gen dataset that has been manually filtered and validated (§3.2).
- **IDK-MRC**: final version of our dataset consisting of Human Filt dataset with additional questions written by annotators in the manual generation stage (§3.3).

Our final IDK-MRC dataset is the largest publicly available Indonesian MRC dataset with various types of unanswerable questions.

## 5 Experiments

We evaluate our IDK-MRC dataset by (1) comparing our automatic QG pipeline with several baselines to measure the performance of our automatic generation method, (2) analyzing the quality and cost of the automatic and manual dataset generation to see whether we can benefit from the additional manual/human-labeled data, and (3) comparing MRC models trained with IDK-MRC and others to validate the effectiveness of our dataset in the downstream task.

### 5.1 Automatic Generation Model Evaluation

We compare our QG model to these methods:

- **TF-IDF**: given an answerable question as the query, unanswerable question is generated by retrieving the most relevant question using TF-IDF features (Pedregosa et al., 2011). The similarity between the questions are calculated using cosine similarity.
- **Rule-based**: we replace the entity in the answerable question with another entity in the context that has the same type, i.e., an entity

with type PERSON will be replaced by another entity with type PERSON. If there is no appropriate entity in the question, we randomly swap the question tag with another tag. We extract the entity using XLM-$R_{BASE}$ model[8] and extract the question tag using a simple matching with our predefined question tag list.

- **Pair2Seq**: we adapt the pair-to-sequence model by Zhu et al. (2019) with Indonesian FastText (Bojanowski et al., 2017). We follow the model architecture and the training procedure described in their paper.

**Dataset** We use SQuAD 2.0 to train our QG model. First, we align answerable and unanswerable questions with the same plausible answer. For example, if there exists an answerable question such as:

> *Who ruled the duchy of Normandy?*
> Answer: ***Richard***

and an unanswerable question such as:

> *Who ruled the country of Normandy?*
> Answer: [empty]
> Plausible answer: ***Richard***

the above questions will be paired or aligned.

Then, we translate the dataset using Google Translate API v2. Because complex questions tend to have more translation artifacts, we eliminate such questions by removing questions with a conjunction. From this process, we get 14,029 input pairs as the training data and 2,144 as the validation data. We use this dataset to train the question generation model (§3.1).

**Implementation** We implement QG model using SimpleTransformers (Rajapakse, 2019). We use $mT5_{BASE}$ (580M parameters) to generate the questions with the maximum sequence length of 512. We train the model in 5 epochs and a batch size of 8. For the decoding, we use top-k and top-p sampling with a value of 50 and 0.95, respectively. We set the returned sequence number to 10.

**Evaluation Metric** We evaluate the models using the existing **BLEU** score metric[9]. However,

[8] https://huggingface.co/cahya/xlm-roberta-base-indonesian-NER

[9] We use the NLTK version of the BLEU score (https://www.nltk.org/api/nltk.translate.bleu_score.html). The tokenization is done using the Indonesian tokenizer of Stanza library (https://stanfordnlp.github.io/stanza).

| Model | UBLEU 3 | UBLEU 4 | BLEU 3 | BLEU 4 | % diff |
|---|---|---|---|---|---|
| TF-IDF | 12.30 | 7.09 | 12.30 | 7.09 | **100%** |
| Rule-based | 41.26 | 33.92 | 41.26 | 33.92 | **100%** |
| Pair2Seq | 26.21 | 19.43 | 28.58 | 21.42 | 94.68% |
| Ours | **43.81** | **36.50** | **43.97** | **36.63** | 99.61% |
| + QA Filter | 42.97 | 35.58 | 43.14 | 35.72 | 99.58% |

Table 3: Automatic evaluation of various QG models tested on translated version of SQuAD 2.0 dev dataset. **%diff**: proportion of generated UnAns questions that is not identical (different) to the paired Ans questions; **BLEU**: BLEU score; **UBLEU**: Unanswerable BLEU.

| Model | UnAns | Rel | Flue | Avg | % prf |
|---|---|---|---|---|---|
| TF-IDF | 0.74 | 2.18 | **2.60** | 0.778 | 20% |
| Rule-based | 0.67 | **2.95** | 2.48 | 0.827 | 40% |
| Pair2Seq | 0.84 | 2.25 | 1.91 | 0.742 | 11% |
| Ours | 0.79 | 2.90 | 2.45 | 0.858 | 50% |
| + QA Filter | **0.89** | 2.92 | 2.48 | **0.897** | **59%** |

Table 4: Human evaluation result from 100 randomly sampled unanswerable questions. **UnAns**: Unanswerability; **Rel**: Relevency; **Flue**: Fluency; **Avg**: Average of UnAns, Rel, Flue, normalized in 0–1 scale; **% prf**: % of samples with perfect UnAns, Rel, and Flue scores.

BLEU is an n-gram based metric, so it can give a high score to the unanswerable questions that are exactly the same as the answerable question. Therefore, we use **%diff** to compute the proportion of generated unanswerable questions that is not identical to its corresponding answerable questions. We also propose a new metric called **Unanswerable BLEU (UBLEU)**, an improved version of BLEU by setting the modified precision ($p_n$) to 0 if the output from the QG model ($q_{out}$) is identical to the paired answerable question ($q_{ans}$), formally defined as:

$$p_n = \frac{\sum_{C \in \{Candidates\}} \sum_{n\text{-}gram \in C} \alpha \; Count_{clip}\,(n\text{-}gram)}{\sum_{C' \in \{Candidates\}} \sum_{n\text{-}gram' \in C'} Count\,(n\text{-}gram')}$$

where

$$\alpha = \begin{cases} 0 & \text{if } q_{out} = q_{ans} \\ 1 & \text{otherwise} \end{cases}$$

Moreover, we conduct human evaluation to further study the performance of the models. We randomly sample 100 questions for each QG models, and ask four annotators to evaluate the questions quality using the same protocol as §3.2.

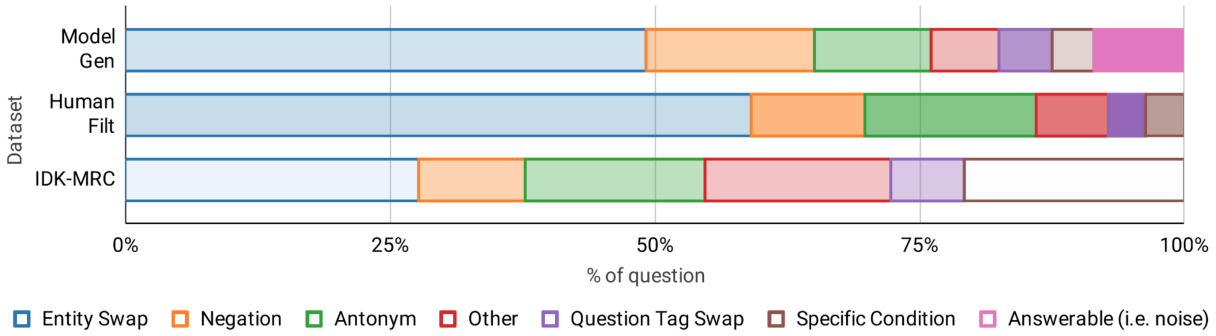


Figure 3: Unanswerable question types distribution of Model Gen, Human Filt, and IDK-MRC test set. The question types are manually labeled by annotators. The bar opacity represents failure rate of the MRC model (XLM-R) in predicting the answer to the questions in each unanswerable question type (lighter is better). Our IDK-MRC dataset has a more balanced question type distribution, resulting in lower failure rate compared to Model Gen and Human Filt dataset.

**Result** As presented in Table 3 and 4, our QG model shows the best performance in both automatic and human evaluation. Despite a lower %diff score than TF-IDF and rule-based, our model still achieves better UBLEU and BLEU scores. We also observe a slight reduction of UBLEU and BLEU scores when we add QA filter; however, based on human evaluation, QA Filter can improve the overall quality of the generated questions, especially the percentage of questions with perfect scores.

TF-IDF has a high fluency score because we use the existing answerable questions from different paragraphs, but it results in a low relevancy score. For rule-based, changing the entity in the answerable question to another entity in the context can produce high relevancy. However, it can still generate an answerable question, as shown by a lower unanswerability score. Pair2Seq (Zhu et al., 2019) obtains a high unanswerability score but lower relevancy and fluency scores. It suffers from many UNK tokens, displaying the limitation of word embedding representation. Overall, adding QA Filter results in better performance in all evaluation aspects, indicated by a high average score and the number of samples with a perfect score.

### 5.2 Automatic vs. Manual Generation

We now compare the automatic and manual dataset generation from three perspectives: time, cost, and question quality, especially to further analyze whether the automatic generation model can benefit from additional human annotation.

**Time and Cost** For the automatic generation, it takes around 3 hours to train QG model on a single RTX 8000 48GB GPU. After the training has finished, the model takes 30 minutes to generate ∼2,000 questions in the inference step. As for the manual process, one person spent 32 hours verifying ∼2,000 questions and 10 hours writing ∼500 questions (40 hours per 2,000 questions). The cost for one human annotator is about $7.5/hour, and assuming a GPU price of $3/hour[10], automatic generation is certainly more time- and cost-efficient approach than manual generation.

**Question Quality** From Figure 3, we observe that our automatic QG model manages to generate various unanswerable question types, as shown in Model Gen question types distribution. However, it still produces some noise, i.e., answerable questions (8.51% of Model Gen test set), even after such questions are discarded by QA Filter model. We also observe that the QG model tends to produce more *entity swap* questions (49.17% of Model Gen test set). Moreover, many irrelevant or incomprehensible questions are still exist in Model Gen dataset, especially for *negation*, *question tag swap*, and *specific condition* types. This result suggests that even though automatic QG model can generate relatively fluent and valid questions, relying *only* on it for building the dataset may result in noise and imbalance question types distribution. Filtering out the noisy data, automatically or manually, is not enough since the question types distribution is still imbalanced, as can be seen in Human Filt question distribution. The additional human-written questions in IDK-MRC cover this limitation, resulting

[10]The highest GPU hourly price from https://cloud.google.com/compute/gpus-pricing (Accessed June 2022).

| Model | Train Dataset | UnAns | | Overall | | Avg UnAns Failure Rate |
|---|---|---|---|---|---|---|
| | | EM | F1 | EM | F1 | % |
| IndoBERT | Translated SQuAD | 61.00 | 61.00 | 52.42 | 59.40 | 45.96 |
| | TyDiQA | 0.19 | 0.19 | 31.00 | 37.08 | 98.47 |
| | Model Gen | 67.44 | 67.44 | 62.09 | 67.45 | 42.42 |
| | Human Filt | 66.64 | 66.64 | 62.49 | 68.19 | 52.35 |
| | IDK-MRC | **86.26** | **86.26** | **72.06** | **77.45** | **31.92** |
| m-BERT | Translated SQuAD | 66.49 | 66.49 | 59.19 | 65.48 | 26.52 |
| | TyDiQA | 0.57 | 0.57 | 36.35 | 41.23 | 99.26 |
| | Model Gen | 79.10 | 79.10 | 72.35 | 77.00 | 19.16 |
| | Human Filt | 69.81 | 69.81 | 68.84 | 73.73 | 38.97 |
| | IDK-MRC | **87.82** | **87.82** | **77.20** | **82.23** | **13.72** |
| XLM-R | Translated SQuAD | 66.78 | 66.78 | 59.00 | 65.89 | 26.13 |
| | TyDiQA | 0.90 | 0.90 | 33.01 | 39.74 | 97.32 |
| | Model Gen | 75.45 | 75.45 | 68.46 | 74.40 | 26.07 |
| | Human Filt | 67.87 | 67.87 | 64.95 | 71.32 | 41.96 |
| | IDK-MRC | **88.29** | **88.29** | **74.86** | **81.37** | **16.42** |

Table 5: MRC models performance trained on various dataset. The EM and F1 scores are the models' performance on IDK-MRC test set, while the Avg Unanswerability Failure Rate are the models' performance on synthetic test cases generated using CheckList tool (Ribeiro et al., 2020). We report average scores over 5 runs. The performance difference between the models trained on our dataset and the baselines are statistically significant ($p < 0.05$).

in a more balanced question type distribution and lower models' failure rate in predicting the answer for each unanswerable question type.

### 5.3 Dataset Evaluation on Downstream Task

Next, we investigate the performance of MRC models trained on our dataset and compare them with Translated SQuAD 2.0 (Muis and Purwarianti, 2020) and TyDiQA (Clark et al., 2020) datasets.

**Implementation** We pick IndoBERT$_{\text{BASE}}$ (Wilie et al., 2020) as the monolingual model and m-BERT$_{\text{BASE}}$ (Devlin et al., 2019), XLM-R$_{\text{BASE}}$ (Conneau et al., 2020) as the multilingual model. They have 124.5M, 167.4M, and 278.7M parameters, respectively. We implemented the models using SimpleTransformers (Rajapakse, 2019). We use the standard hyperparameter settings for QA task with maximum sequence length of 384, document stride of 128, and trained the models for 10 epochs, batch size of 8, learning rate of 2e-5 using the Adam optimizer (Kingma and Ba, 2014).

**Result on IDK-MRC test set** We tested several models using IDK-MRC test set[11] and the result is presented in Table 5. We observe that the models trained on IDK-MRC dataset perform better than all baselines. Furthermore, even the models trained on our less-cleaned Model Gen dataset can obtain a better result than Translated SQuAD dataset, indicating the importance of a dataset that originates in Indonesian. We also note that the models trained on TyDiQA fail to handle unanswerable questions, as shown by extremely low UnAns scores. This result further highlights the significance of incorporating unanswerable questions in MRC dataset.

**Unanswerable failure rate** To further examine the models' capability in handling unanswerable questions, we also conduct an unanswerability error analysis using CheckList (Ribeiro et al., 2020), a tool that facilitates behavioral test on many NLP tasks. CheckList provides a list of linguistic capabilities, with each capabilities are divided into different *test types* to further break down the potential *failure* of the linguistic capabilities. In this experiment, we focus on testing the models' capability on predicting the answer to unanswerable questions by dividing the *test types* into the unan-

[11] We use IDK-MRC test set since there is no suitable existing dataset to show the performance on Indonesian unanswerable questions. Another option is to test the models on Translated SQuAD 2.0 test set, but we found that many questions have machine translation error or incomplete or wrong ground truth answer (43% out of 100 randomly sampled questions). Therefore, Translated SQuAD 2.0 is not an adequate dataset to test Indonesian MRC models.

swerable questions type listed in Table 1. The test cases for each test type are automatically generated using CheckList's `template` function, resulting in 600 test cases for each question type. The template examples are presented in Appendix B.

As shown in last column of Table 5, we find that our dataset can reduce the average failure rate, further confirming the effectiveness of IDK-MRC compared to the existing dataset. IndoBERT has the most significant failure rate reduction (45.96 to 31.92), followed by m-BERT (26.52 to 13.72) and XLM-R (26.13 to 16.42). Also, it is clear that only relying on the existing TyDiQA dataset is not enough to build a robust model, as shown from very high failure rates.

Overall, most failures occur on *negation* questions, specifically when the negation word appears in the context passage, such as:

> Context: *Wikia **tidak** diketuai oleh Ali.*
> *(Wikia **is not** chaired by Ali.)*
> Question: *Siapa yang mengetuai Wikia?*
> *(Who is Wikia's chair?)*
> Predicted Answer: *Ali*
> Correct Answer: `[empty]`

Besides adding more data samples, we conjecture that some improvement in model architecture or training scheme is needed to solve this problem. It is possible that the model highly correlates "who" question tag with any person's name that appears in the context, and picks it as the answer without considering the meaning of the whole context. Additionally, the high failure rates on IndoBERT model are mainly contributed by the *antonym* and *question tag swap* types, while multilingual models like m-BERT and XLM-R performs significantly better on this type of question. All in all, focusing on better approach to handle the aforementioned question types for future work may further improve the models' performance.

## 6 Conclusion

We have presented IDK-MRC, the first Indonesian MRC dataset covering answerable and unanswerable questions. We confirm the effectiveness of our dataset in improving the MRC models' capability to handle unanswerable questions compared to other existing MRC datasets, such as Translated SQuAD and TyDiQA. We also verify that our automatic dataset generation method can help reduce the time and cost of the dataset collection. Subsequently, human supervision helps eliminate the dataset noise and question type imbalance problem from the automatic generation method.

Although our dataset collection pipeline is designed to build unanswerable questions for Indonesian, it can also be utilized for other medium- to low-resource languages or other QA tasks, such as adversarial question generation. While our dataset pipeline (i.e., automatic generation, validation, manual generation) is general enough to be applied to other languages or QA tasks, further adjustment of the automatic question generation model is required. Still, we believe that our proposed pipeline has some potential to be generalized to several QA tasks, which may be an interesting direction for future work.

## Limitations

There may be some possible limitations in our study. Firstly, our automatic question generation (QG) model requires training data consisting of context paragraphs and answerable questions. Unlike medium- to low-resource languages like Indonesian, our QG method might be more challenging to be applied to extremely low-resource languages with even more limited data and resources.

Secondly, we utilized the existing transformer-based models that specifically pre-trained on Indonesian language, i.e., IndoBERT (Wilie et al., 2020). While we also used multilingual models like mT5 (Xue et al., 2021), m-BERT (Devlin et al., 2019), and XLM-R (Conneau et al., 2020), the number of languages covered by these models is also limited. Before applying those models to other language besides Indonesian, one must check whether the desired language exists during the pre-training phase of the models. Note that the model also needs to have high enough quality. Some of the large multilingual models are not very good for low- to extremely low-resource languages.

## Ethics Statement

The paragraphs and answerable questions that we utilized to build IDK-MRC dataset are taken from Indonesian subset of TyDiQA-GoldP dataset (Clark et al., 2020), which originates from Wikipedia articles. Since those articles are written from a neutral point of view, the risk of harmful content is minimal. Also, all model-generated questions in our dataset have been validated by human annotators to

eliminate the risk of harmful questions. During the manual question generation process, the annotators are also encouraged to avoid producing possibly offensive questions.

Even so, we argue that further assessment is needed before using our dataset and models in real-world applications. This measurement is especially required for the pre-trained language models used in our experiments, namely mT5 (Xue et al., 2021), IndoBERT (Wilie et al., 2020), m-BERT (Devlin et al., 2019), and XLM-R (Conneau et al., 2020). These language models are mostly pre-trained on the common-crawl dataset, which may contain harmful biases or stereotypes.

All datasets in this work are publicly available and distributed under CC BY-SA 4.0 license. Our data collection pipeline, along with the recruitment process of the human annotators, has been reviewed and approved by KAIST Institutional Review Board (KH2021-194). We ensured that annotators were paid above the minimum wage in the Republic of Korea.

## Acknowledgements

This work was partly supported by Institute of Information & communications Technology Planning & Evaluation (IITP) grant funded by the Korea government(MSIT) (No. 2022-0-00184, Development and Study of AI Technologies to Inexpensively Conform to Evolving Policy on Ethics). We also would like to thank Dea Adhista for managing the annotators during the validation and manual data collection process. Rifki Afina Putri was supported by Hyundai Motor Chung Mong-Koo Global Scholarship.

## References

Chris Alberti, Daniel Andor, Emily Pitler, Jacob Devlin, and Michael Collins. 2019. Synthetic QA corpora generation with roundtrip consistency. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 6168–6173, Florence, Italy. Association for Computational Linguistics.

Max Bartolo, Alastair Roberts, Johannes Welbl, Sebastian Riedel, and Pontus Stenetorp. 2020. Beat the AI: Investigating adversarial human annotation for reading comprehension. *Transactions of the Association for Computational Linguistics*, 8:662–678.

Max Bartolo, Tristan Thrush, Robin Jia, Sebastian Riedel, Pontus Stenetorp, and Douwe Kiela. 2021. Improving question answering model robustness with synthetic adversarial data generation. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 8830–8848, Online and Punta Cana, Dominican Republic. Association for Computational Linguistics.

Piotr Bojanowski, Edouard Grave, Armand Joulin, and Tomas Mikolov. 2017. Enriching word vectors with subword information. *Transactions of the Association for Computational Linguistics*, 5:135–146.

Tom Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared D Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, Sandhini Agarwal, Ariel Herbert-Voss, Gretchen Krueger, Tom Henighan, Rewon Child, Aditya Ramesh, Daniel Ziegler, Jeffrey Wu, Clemens Winter, Chris Hesse, Mark Chen, Eric Sigler, Mateusz Litwin, Scott Gray, Benjamin Chess, Jack Clark, Christopher Berner, Sam McCandlish, Alec Radford, Ilya Sutskever, and Dario Amodei. 2020. Language models are few-shot learners. In *Advances in Neural Information Processing Systems*, volume 33, pages 1877–1901. Curran Associates, Inc.

Zewen Chi, Li Dong, Furu Wei, Wenhui Wang, Xian-Ling Mao, and Heyan Huang. 2020. Cross-lingual natural language generation via pre-training. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 34, pages 7570–7577.

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. *Transactions of the Association for Computational Linguistics*, 8:454–470.

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 *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 8440–8451, Online. Association for Computational Linguistics.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. 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)*, pages 4171–4186, Minneapolis, Minnesota. Association for Computational Linguistics.

Xinya Du and Claire Cardie. 2018. Harvesting paragraph-level question-answer pairs from Wikipedia. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1907–1917, Melbourne, Australia. Association for Computational Linguistics.

Xinya Du, Junru Shao, and Claire Cardie. 2017. Learning to ask: Neural question generation for reading comprehension. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1342–1352, Vancouver, Canada. Association for Computational Linguistics.

Michael Heilman and Noah A. Smith. 2010. Good question! statistical ranking for question generation. In *Human Language Technologies: The 2010 Annual Conference of the North American Chapter of the Association for Computational Linguistics*, pages 609–617, Los Angeles, California. Association for Computational Linguistics.

Sepp Hochreiter and Jürgen Schmidhuber. 1997. Long short-term memory. *Neural computation*, 9(8):1735–1780.

Diederik P Kingma and Jimmy Ba. 2014. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*.

Tassilo Klein and Moin Nabi. 2019. Learning to answer by learning to ask: Getting the best of gpt-2 and bert worlds. *arXiv preprint arXiv:1911.02365*.

Vishwajeet Kumar, Nitish Joshi, Arijit Mukherjee, Ganesh Ramakrishnan, and Preethi Jyothi. 2019. Cross-lingual training for automatic question generation. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 4863–4872, Florence, Italy. Association for Computational Linguistics.

Patrick Lewis, Ludovic Denoyer, and Sebastian Riedel. 2019. Unsupervised question answering by cloze translation. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 4896–4910, Florence, Italy. Association for Computational Linguistics.

Yaobo Liang, Nan Duan, Yeyun Gong, Ning Wu, Fenfei Guo, Weizhen Qi, Ming Gong, Linjun Shou, Daxin Jiang, Guihong Cao, Xiaodong Fan, Ruofei Zhang, Rahul Agrawal, Edward Cui, Sining Wei, Taroon Bharti, Ying Qiao, Jiun-Hung Chen, Winnie Wu, Shuguang Liu, Fan Yang, Daniel Campos, Rangan Majumder, and Ming Zhou. 2020. XGLUE: A new benchmark datasetfor cross-lingual pre-training, understanding and generation. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 6008–6018, Online. Association for Computational Linguistics.

Ferdiant Joshua Muis and Ayu Purwarianti. 2020. Sequence-to-sequence learning for indonesian automatic question generator. In *2020 7th International Conference on Advance Informatics: Concepts, Theory and Applications (ICAICTA)*, pages 1–6.

Yixin Nie, Adina Williams, Emily Dinan, Mohit Bansal, Jason Weston, and Douwe Kiela. 2020. Adversarial NLI: A new benchmark for natural language understanding. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 4885–4901, Online. Association for Computational Linguistics.

F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, J. Vanderplas, A. Passos, D. Cournapeau, M. Brucher, M. Perrot, and E. Duchesnay. 2011. Scikit-learn: Machine learning in Python. *Journal of Machine Learning Research*, 12:2825–2830.

Jeffrey Pennington, Richard Socher, and Christopher Manning. 2014. GloVe: Global vectors for word representation. In *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 1532–1543, Doha, Qatar. Association for Computational Linguistics.

Raul Puri, Ryan Spring, Mohammad Shoeybi, Mostofa Patwary, and Bryan Catanzaro. 2020. Training question answering models from synthetic data. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 5811–5826, Online. Association for Computational Linguistics.

Ayu Purwarianti, Masatoshi Tsuchiya, and Seiichi Nakagawa. 2007. A machine learning approach for indonesian question answering system. In *Proceedings of the 25th Conference on Proceedings of the 25th IASTED International Multi-Conference: Artificial Intelligence and Applications*, AIAP'07, page 537–542, USA. ACTA Press.

T. C. Rajapakse. 2019. Simple transformers. `https://github.com/ThilinaRajapakse/simpletransformers`.

Pranav Rajpurkar, Robin Jia, and Percy Liang. 2018. 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)*, pages 784–789, Melbourne, Australia. Association for Computational Linguistics.

Pranav Rajpurkar, Jian Zhang, Konstantin Lopyrev, and Percy Liang. 2016. SQuAD: 100,000+ questions for machine comprehension of text. In *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing*, pages 2383–2392, Austin, Texas. Association for Computational Linguistics.

Arij Riabi, Thomas Scialom, Rachel Keraron, Benoît Sagot, Djamé Seddah, and Jacopo Staiano. 2021. Synthetic data augmentation for zero-shot cross-lingual question answering. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 7016–7030, Online and Punta Cana, Dominican Republic. Association for Computational Linguistics.

Marco Tulio Ribeiro, Tongshuang Wu, Carlos Guestrin, and Sameer Singh. 2020. Beyond accuracy: Behavioral testing of NLP models with CheckList. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 4902–4912, Online. Association for Computational Linguistics.

Priyanka Sen and Amir Saffari. 2020. What do models learn from question answering datasets? In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 2429–2438, Online. Association for Computational Linguistics.

Siamak Shakeri, Noah Constant, Mihir Kale, and Linting Xue. 2021. Towards zero-shot multilingual synthetic question and answer generation for cross-lingual reading comprehension. In *Proceedings of the 14th International Conference on Natural Language Generation*, pages 35–45, Aberdeen, Scotland, UK. Association for Computational Linguistics.

Siamak Shakeri, Cicero Nogueira dos Santos, Henghui Zhu, Patrick Ng, Feng Nan, Zhiguo Wang, Ramesh Nallapati, and Bing Xiang. 2020. End-to-end synthetic data generation for domain adaptation of question answering systems. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 5445–5460, Online. Association for Computational Linguistics.

Adam Trischler, Tong Wang, Xingdi Yuan, Justin Harris, Alessandro Sordoni, Philip Bachman, and Kaheer Suleman. 2017. NewsQA: A machine comprehension dataset. In *Proceedings of the 2nd Workshop on Representation Learning for NLP*, pages 191–200, Vancouver, Canada. Association for Computational Linguistics.

Shuohang Wang, Yang Liu, Yichong Xu, Chenguang Zhu, and Michael Zeng. 2021. Want to reduce labeling cost? GPT-3 can help. In *Findings of the Association for Computational Linguistics: EMNLP 2021*, pages 4195–4205, Punta Cana, Dominican Republic. Association for Computational Linguistics.

Bryan Wilie, Karissa Vincentio, Genta Indra Winata, Samuel Cahyawijaya, Xiaohong Li, Zhi Yuan Lim, Sidik Soleman, Rahmad Mahendra, Pascale Fung, Syafri Bahar, and Ayu Purwarianti. 2020. IndoNLU: Benchmark and resources for evaluating Indonesian natural language understanding. In *Proceedings of the 1st Conference of the Asia-Pacific Chapter of the Association for Computational Linguistics and the 10th International Joint Conference on Natural Language Processing*, pages 843–857, Suzhou, China. Association for Computational Linguistics.

Thomas Wolf, Lysandre Debut, Victor Sanh, Julien Chaumond, Clement Delangue, Anthony Moi, Pierric Cistac, Tim Rault, Rémi Louf, Morgan Funtowicz, Joe Davison, Sam Shleifer, Patrick von Platen, Clara Ma, Yacine Jernite, Julien Plu, Canwen Xu, Teven Le Scao, Sylvain Gugger, Mariama Drame, Quentin Lhoest, and Alexander M. Rush. 2020. Transformers: State-of-the-art natural language processing. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing: System Demonstrations*, pages 38–45, Online. Association for Computational Linguistics.

Linting Xue, Noah Constant, Adam Roberts, Mihir Kale, Rami Al-Rfou, Aditya Siddhant, Aditya Barua, and Colin Raffel. 2021. mT5: A massively multilingual pre-trained text-to-text transformer. In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 483–498, Online. Association for Computational Linguistics.

Haichao Zhu, Li Dong, Furu Wei, Wenhui Wang, Bing Qin, and Ting Liu. 2019. Learning to ask unanswerable questions for machine reading comprehension. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 4238–4248, Florence, Italy. Association for Computational Linguistics.

# Appendix for "IDK-MRC: Unanswerable Questions for Indonesian Machine Reading Comprehension"

## A Data Statement

### A.1 Curation Rationale

IDK-MRC dataset is built based on the existing paragraph and answerable questions (`ans`) in TyDiQA-GoldP (Clark et al., 2020). The new unanswerable questions are automatically generated using the combination of mT5 (Xue et al., 2021) and XLM-R (Conneau et al., 2020) models, which are then manually verified by human annotators (`filtered ans` and `filtered unans`). We also asked the annotators to manually write additional unanswerable questions as described in §3.3 (`additional unans`). Each paragraphs in the final dataset will have a set of `filtered ans`, `filtered unans`, and `additional unans` questions. The illustration of the dataset collection pipeline is shown in Figure 1.

### A.2 Language Variety

The texts in IDK-MRC are generated and written using the standard formal style of the Indonesian language.

### A.3 Annotator Demographic

In our dataset collection pipeline, the annotators are asked to validate the generated unanswerable questions and write a new additional unanswerable questions.

We recruit four annotators with 2+ years of experience in Indonesian NLP annotation using direct recruitment. All of them are Indonesian native speakers who reside in Indonesia (Java Island) and fall under the 18–34 age category. We set the payment to around $7.5 per hour. Given the annotators' demographic, we ensure that the payment is above the minimum wage rate (as of December 2021). All annotators also have signed the consent form and agreed to participate in this project.

### A.4 Speech Situation

The paragraphs and answerable questions in IDK-MRC are built based on TyDiQA-GoldP (Clark et al., 2020), which was originally taken from 2019 snapshots of Indonesian Wikipedia. As for the unanswerable questions, the dataset collection is conducted in December 2021. However, it is generated or written based on the facts or information provided in the existing paragraph in TyDiQA-GoldP dataset.

| Question Type | Template Example |
|---|---|
| Negation (in question) | A is `VERB` by B. Who is not `VERB` B? |
| Negation (in context) | A is not `VERB` by B. Who is `VERB` B? |
| Antonym | A got the `ADJ` prize. Who got the `antonym of ADJ` prize? |
| Entity Swap | A is the president of B. Who is the president of C? |
| Question Tag Swap | A was found on `DATE`. Who found A? |
| Specific Condition | A is the president of B. Who is the (first) president of B? |
| Other | A is `NOUN1` of B. Who is `NOUN2` of B? |

Table 6: The test case template examples for 'who' question tag.

### A.5 Text Characteristics

The original texts in IDK-MRC are mainly based from Wikipedia articles covering various topics, such as history, science, biography, and many more[12].

### A.6 Provenance Appendix

As described in the previous section, the paragraphs and answerable questions in IDK-MRC are taken from the existing Indonesian TyDiQA dataset (Clark et al., 2020). Unfortunately, the authors of TyDiQA did not provide complete data statement information, especially on their annotators demographic. However, since the original source of this dataset is from Wikipedia, we conjecture that the speech situation and text characteristic of this dataset is not far from the one that we have discussed in previous sections.

## B Unanswerability Analysis by Question Type

In this section, we aim to further measure the MRC models' performance on handling each unanswerable question type using CheckList tool.

[12]The complete list of Indonesian Wikipedia article topics can be seen in https://id.wikipedia.org/wiki/Wikipedia:Artikel_pilihan/Topik

| | Failure Rate | | | | | | | | | | | | | | | Failure Cases Examples with expected answer (A) and model prediction (P) |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | IndoBERT | | | | | m-BERT | | | | | XLM-R | | | | | |
| | SQ | TY | MG | HF | IDK | SQ | TY | MG | HF | IDK | SQ | TY | MG | HF | IDK | |
| Negation | 0.3 | 99.4 | **0.0** | 0.4 | 0.9 | 0.1 | 99.6 | **0.0** | **0.0** | 1.1 | **0.0** | 95.2 | **0.0** | **0.0** | **0.0** | C: Wikia dirancang oleh James.<br>*Wikia was designed by James.*<br>Q: Apa yg tidak dirancang James?<br>*What was not designed by James?*<br>A: `[empty]` P: Wikia |
| | **35.0** | 100 | 66.8 | 83.5 | 60.9 | **4.4** | 99.5 | 55.5 | 84.1 | 19.4 | **16.3** | 94.6 | 53.9 | 77.8 | 49.6 | C: Wikia tidak diketuai oleh Ali.<br>*Wikia is not chaired by Ali.*<br>Q: Siapa yg mengetuai Wikia?<br>*Who is Wikia's chair?*<br>A: `[empty]` P: Ali |
| Antonym | **31.2** | 100 | 63.9 | 88.7 | 74.4 | **10.3** | 99.9 | 20.7 | 45.6 | 16.8 | **7.2** | 99.3 | 24.9 | 41.0 | 22.5 | C: Bia mendapatkan hadiah terendah.<br>*Bia got the lowest prize.*<br>Q: Siapa yg dapat hadiah tertinggi?<br>*Who got the highest prize?*<br>A: `[empty]` P: Bia |
| Ent Swap | 46.7 | 98.0 | 19.1 | 14.7 | **14.2** | 9.1 | 100 | **0.8** | 2.3 | 2.0 | 30.7 | 99.5 | 15.0 | 17.3 | **5.9** | C: Dewi adalah presiden Kolombia.<br>*Dewi is the president of Colombia.*<br>Q: Siapa presiden Chili?<br>*Who is the president of Chile?*<br>A: `[empty]` P: Dewi |
| QTag Swap | **48.8** | 95.5 | 75.2 | 86.8 | 50.2 | 35.4 | 96.2 | 49.3 | 72.5 | **27.4** | **20.5** | 91.8 | 48.7 | 69.5 | 24.9 | C: Anita lahir di Israel.<br>*Anita was born in Israel.*<br>Q: Kapan Anita lahir?<br>*When was Anita born?*<br>A: `[empty]` P: Israel |
| Specific Cond. | 74.6 | 99.9 | 26.4 | 38.0 | **4.7** | 75.0 | 100 | 6.5 | 37.7 | **5.7** | 62.8 | 100 | 18.2 | 40.8 | **0.7** | C: Roy adalah seorang presiden.<br>*Roy is a president.*<br>Q: Siapa presiden paling terkenal?<br>*Who is the most famous president?*<br>A: `[empty]` P: Roy |
| Other | 56.9 | 97.7 | 36.6 | 44.0 | **17.0** | 27.1 | 99.9 | **10.0** | 33.6 | 20.3 | 27.5 | 98.5 | 22.7 | 44.3 | **19.8** | C: Sheila adalah penggemar Rudy.<br>*Sheila is Rudy's fan.*<br>Q: Siapa teman Rudy?<br>*Who is Rudy's friend?*<br>A: `[empty]` P: Sheila |
| | 45.96 | 98.47 | 42.42 | 52.35 | **31.92** | 26.52 | 99.26 | 19.16 | 38.97 | **13.72** | 26.13 | 97.32 | 26.07 | 41.96 | **16.42** | |

Table 7: The failure rate on all unanswerable question types tested using the CheckList tool (Ribeiro et al., 2020). The scores are the mean over 5 runs with different random seeds (lower score is better). The **last row** denotes the **macro average** of all unanswerability types. **SQ**: translated SQuAD, **TY**: TyDiQA, **MG**: Model Gen, **HF**: Human Filt, **IDK**: IDK-MRC.

**Test Case Generation** We utilized `template` function provided in Checklist to generate the test cases for each unanswerable question type, i.e., negation (in-question and in-context), antonym, entity swap, question tag swap, specific condition, and other. Each question type consists of several question tag, namely *siapa* (who), *apa* (what), *kapan* (when), *di mana* (where), *mengapa* (why), and *berapa* (how long/many/much). Each question tag have 100 test cases, therefore, we have a total of 600 test cases for each unanswerable question type. Some of the template examples are presented in Table 6.

**Experiment Result** As shown from Table 7, models trained on our IDK-MRC dataset has a lower failure rate on the *entity swap*, *question tag swap*, *specific condition*, and *other* questions, indicating that adding more examples for these question types can improve the models' unanswerability skills. Also, most models can successfully handle the negation if the negated word exists in the question. When the negated word appears in the context, most models fail to predict the correct answer. Meanwhile, models train on SQuAD has a lowest failure rate on negation case, and we conjecture that it occurs due to the imbalanced question type distribution in the SQuAD training dataset. As reported by Sen and Saffari (2020), 85% of questions containing `"n't"` and 89% of questions containing `"never"` in SQuAD dataset are categorized as unanswerable question. It aligns with our experiment results, which shows that SQuAD has a lowest failure rate on negation question type and a much higher failure rate on the other question types.

**TASK 1**

In this task, you are asked to evaluate the quality of the unanswerable question that has been automatically generated a Question Generation model.
The unanswerable questions are generated from TyDiQA data, which has 5369 training data, 402 validation data, and 423 testing data.
Each person has to annotate a total of ±1548 questions and evaluate another questions that have been annotated by another annotator.

For each unanswerable question, the paragraph and the answerable question will be given as a reference.
You have to **evaluate** the **unanswerable questions** based on three different criteria:

1. **Unanswerability (UnAns)**

   Can we **found the answer** on the given paragraph?

| Score | Interpretation |
|---|---|
| **1** | The answer **cannot** be found on the given paragraph |
| **0** | The answer **can** be found on the given paragraph |

2. **Relevancy**

   Is the question **relevant** to the **paragraph and the answerable question**?

| Score | Interpretation |
|---|---|
| **3** | **Relevant** to **both** |
| **2** | Only relevant to the paragraph **or** answerable question |
| **1** | **Not relevant** to **both** |

3. **Fluency**

   How **fluent** the question is?

| Score | Interpretation |
|---|---|
| **3** | Collective quality of all words in question is fluent and coherent |
| **2** | Minor typo and/or grammatical errors; semicoherent |
| **1** | Incomprehensible; incoherent |

**Additional Rules from Discussion**

1. The slightest writing error in the entity will be immediately considered as different entity.
2. A definition question that falls under negation question type have a fluency of 1.
3. The minor fix for question with scores 1, 3, 2 shoud be unanswerable.
4. Relevancy: one entity cluster (relevant), one term cluster (irrelevant), unless the term is contained in the paragraph.
5. The fluency score for the unans Q in the form of a statement sentence (ends with a punctuation mark) is 1 because it does not meet with the requirements of a question sentence.
6. The fluency score for yes/no question is 1 (out of dataset scope).

Figure 4: Annotation instruction for the validation stage.

# C Annotation Instruction

## C.1 Validation Stage

In this stage, human annotators are asked to validate the quality of the model-generated unanswerable questions using criteria as described in §3.2. Detailed instruction can be seen in Figure 4.

## C.2 Manual Generation Stage

In this stage, human annotators are asked to write a questions that the question generation model fails to generate as described in §3.3. The instruction given to the annotators are shown in Figure 5.

**TASK 2**

In this task, you are asked to add additional unanswerable questions, especially to the paragraphs that don't have a valid unanswerable question yet.

There are several unanswerable question types as follows:

| Type | Description | Example | Current Test | |
|---|---|---|---|---|
| | | | Num | % |
| Negation | Negation word inserted or removed | *Paragraph* Kambing **tidak** memiliki lemak dalam kandungan susunya.<br>*Ans Q* Apakah kandungan yang **tidak** **ada** dalam susu kambing?<br>*UnAns Q* Apakah kandungan yang **ada** dalam susu kambing? | 27 | 10.84% |
| | | *Paragraph* Membran adalah kulit **tipis** yang berfungsi sebagai pemisah selektif.<br>*Ans Q* Kulit **tidak** **tebal** yang berfungsi sebagai pemisah selektif disebut apa?<br>*UnAns Q* Kulit **tebal** yang berfungsi sebagai pemisah selektif disebut apa? | | |
| Antonym | Antonym used | *Paragraph* Aristokrasi adalah sebuah kelas sosial yang dalam sebagian besar tatanan sosial dianggap yang tertinggi di kalangan masyarakat.<br>*Ans Q* Apakah nama kelas sosial tertinggi?<br>*UnAns Q* Apa nama kelas sosial terendah? | 40 | 16.06% |
| Entity Swap | Entity or term replaced with other entity or term | *Paragraph* Terdapat dua kandidat standar untuk 4G yang dikomersilkan di dunia yaitu standar WiMAX (Korea Selatan sejak 2006) dan standar Long Term Evolution (LTE) (Swedia sejak 2009).<br>*Ans Q* Di manakah LTE pertama kali diciptakan?<br>*UnAns Q* Di manakah 3G pertama diciptakan? | 147 | 59.04% |
| Question Tag Swap | Question tag replaced with other question tag | *Paragraph* Suaka margasatwa Muara Angke (SMMA) adalah sebuah kawasan konservasi berdasarkan SK Menteri Kehutanan RI No. 097/Kpts-II/1988, 29 Februari 1998 di wilayah hutan bakau (mangrove) di pesisir utara Jakarta.<br>*Ans Q* Di mana Suaka margasatwa Muara Angke dibangun?<br>*UnAns Q* Kapan Suaka margasatwa Muara Angke dibangun? | 9 | 3.61% |
| Specific Condition | Asks for condition that is not satisfied by anything in the paragraph | *Paragraph* Bon Jovi saat ini terdiri dari Vokalis Jon Bon Jovi, Keyboardist David Bryan, Drummer Tico Torres, Gitaris Phil X, dan Bassist Hugh McDonald.<br>*Ans Q* Siapa nama personil Bon Jovi?<br>*UnAns Q* Siapa nama personil Bon Jovi yang paling jarang dikenal? | 9 | 3.61% |
| Other | Other cases where the paragraph does not imply any answer | *Paragraph* Patrick Star adalah seekor bintang laut berwarna merah muda yang merupakan sahabat Spongebob.<br>*Ans Q* Siapakah teman baik karakter SpongeBob SquarePants?<br>*UnAns Q* Siapa teman kecil karakter Spongebob SquarePants? | 17 | 6.83% |

**Additional Rules from Discussion**

1. Other: please make the question as close as possible to the corresponding answerable question.
2. Negation: please add the ans Q & unans Q
   a. Additional ans Q should contain negated word.
   b. Additional unans Q is the negation of ans Q.
3. The additional question should meet the 1, 3, 3 score criteria for unans Q, and 0, 3, 3 for ans Q.

Figure 5: Annotation instruction for the manual generation stage.