Post-Training with Interrogative Sentences for Enhancing BART-based Korean Question Generator

Gyu-Min Park[1], **Seong-Eun Hong**[2], **Seong-Bae Park**[1]
[1]School of Computer Science and Engineering
[2]Department of Software Convergence
Kyung Hee University, Korea
{pgm1219,zen152,sbpark71}@khu.ac.kr

Abstract

Pre-trained language models such as KoBART often fail to generate perfect interrogative sentences when they are applied to Korean question generation. This is mainly due to the fact that the language models are trained with declarative sentences, but not with interrogative sentences. Therefore, this paper proposes a novel post-training of KoBART to enhance it for Korean question generation. The enhancement of KoBART is accomplished in three ways: (i) introduction of *question infilling* objective to KoBART to enforce it to focus more on the structure of interrogative sentences, (ii) augmentation of training data for question generation with another MRC data from AI-Hub to cope with the lack of training instances for post-training, (iii) introduction of *Korean spacing* objective to make KoBART understand the linguistic features of Korean. Since there is no standard data set for Korean question generation, this paper also proposes KorQuAD-QG, a new data set for this task, to verify the performance of the proposed post-training. Our code are publicly available at https://github.com/gminipark/post_training_qg.

1 Introduction

Question generation is a task that aims to generate a question automatically from a given context text. Since it is a kind of text generation task, it has wide applications. For instance, it has been used for constructing robust question answering systems (Duan et al., 2017; Le Berre et al., 2022), augmenting data for machine reading comprehension (MRC) (Du et al., 2017; Ghanem et al., 2022), and making goal-oriented dialogue systems (Laban et al., 2020; Gu et al., 2021).

The main approach of question generation is to adopt a pre-trained language model trained with a large-scale corpus and then fine-tune the model with a data set for question generation (Chan and Fan, 2019; Dong et al., 2019; Xiao et al., 2020). In answer-aware question generation, it is important to figure out which part of a content is most relevant and understand the structure of interrogative sentences. However, most current pre-trained language models are not much experienced with the domain of question generation and interrogative sentences. As a result, even the fine-tuned model does not reflect the characteristics of question generation fully.

One solution to this problem is to enforce a language model to contain proper knowledge for question generation. Sun et al. (2021) proposed a language model trained with a knowledge graph and plain texts to make the language model knowledge-enhanced. However, this approach requires a lot of resources to train such a language model since the language model usually has more parameters than ordinary language models. On the other hand, Wang et al. (2021) added an adapter to a pre-trained language model, and only the adapter is trained to capture some knowledge for question generation. However, this approach requires external knowledge for question generation which is difficult to obtain.

Another solution is to adopt the idea of post-training (Gururangan et al., 2020) which adapts a language model to a new task by making the language model learn the objective of the new task or augmenting its training data with those of the task. For instance, Whang et al. (2020) and Han et al. (2021) showed that BERT could be improved in dialogue response selection by learning, as post-training, dialogue data which BERT did not experience in the pre-training step. Many previous studies proved that post-training enhances a pre-trained language model in several classification and text generation tasks (Xu et al., 2019; Whang et al., 2020; Peng et al., 2021), but there is no study that a pre-trained language model improves question generation through post-training with well-designed objectives.

Proceedings of the 2nd Conference of the Asia-Pacific Chapter of the Association for Computational Linguistics and the 12th International Joint Conference on Natural Language Processing (Volume 2: Short Papers), pages 202–209

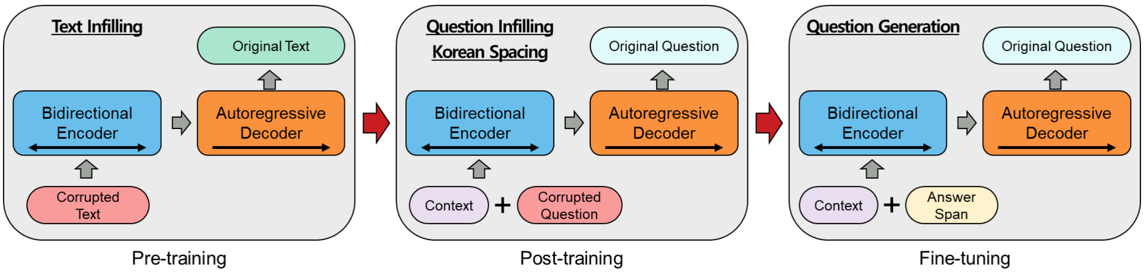

Figure 1: Overview of training the proposed KoBART-based question generator.

This paper proposes a novel post-training of KoBART, a Korean BART, for Korean question generation. The proposed post-training tackles four issues about post-training a BART-based Korean question generator. First, a new data set, KorQuAD-QG, is developed following the work of Lim et al. (2019), since there is no public data set for Korean question generation. Note that KoBART reveals a weakness in generating interrogative sentences since it never experienced the question generation task in its pre-training step. Thus, the proposed post-training adapts KoBART to question generation by enforcing it to focus more on questions with a new objective *question infilling*.

The performance of pre-trained language models is affected by the number of training instances. Thus, KoBART is post-trained with external MRC data as well as KorQuAD-QG. The last issue is related with Korean language. KoBART is missing some linguistic characteristics of Korean interrogative sentences. Therefore, the proposed post-training injects the characteristics explicitly to KoBART by introducing a new objective *Korean spacing*.

2 Related Work

Recent previous studies have shown that large-scale pre-trained language models show prominent performance in many NLP tasks including question generation (Chan and Fan, 2019; Dong et al., 2019; Xiao et al., 2020). For instance, Dong et al. (2019) proposed a unified language model for solving various NLP tasks. For this, they contrived three language modeling objectives of unidirectional objective, bidirectional objective, and seq-to-seq objective, and then applied all the objectives to language modeling. On the other hand, ERINE-GEN achieved the SOTA performance by applying an infilling generation mechanism and a noise-aware generation method to the multi-flow attention architecture (Xiao et al., 2020). However, these language models share a problem that plenty of resources are needed to train them. In addition, they suffer from a lack of domain knowledge of question generation task since they did not experience the sentences for question generation in their pre-training.

One solution to these problems is to post-train a language model before fine-tuning. Post-training of a language model has shown a great performance in many NLP tasks (Gururangan et al., 2020; Whang et al., 2020; Han et al., 2021). Whang et al. (2020) proposed a post-training for response selection which optimizes BERT with the next sentence prediction (NSP) and masked language model (MLM) using the corpus of response selection and then finetines it with the objective of response selection. On the other hand, Han et al. (2021) replaced NSP of BERT with utterance relevance classification (URC) that is more relevant to response selection. They reported that the use of URC instead of NSP led to performance improvement.

3 Korean Question Generation

Question generation is a task of generating a question q from a context C and an answer span A within the context. Thus, a question generator produces an interrogative sentence that maximizes

$$P(q|C, A, \theta) = \prod_{j=1}^{|q|} P\left(q_j|C, A, q_{<j-1}; \theta\right)$$

where θ is a model parameter of the generator.

This paper adopts KoBART[1], a Korean BART, for $P(\cdot)$. BART is a denoising autoencoder which reconstructs an original text from a corrupted text. It is optimized by minimizing the negative log likelihood

$$\mathcal{L}_{pre} = -\sum_{t \in \mathcal{D}} \log P\left(t|t^c; \theta\right), \qquad (1)$$

[1] https://github.com/SKT-AI/KoBART

where $\mathcal{D}$ is a corpus for training BART, t is an original text, and t^c is a corrupted text of t by a transformation method. Token masking, token deletion, text infilling, sentence permutation, and document rotation were proposed as a transformation method, but *text infilling* has shown the best performance in many NLP tasks (Lewis et al., 2020). Thus, KoBART is pre-trained with text infilling.

The pre-trained KoBART is adapted to question generation by fine-tuning the parameter θ with a data set for question generation, $D_{qg} = \{(C_i, A_i, q_i)\}_{i=1}^{N}$. That is, θ is tuned with D_{qg} to minimize

$$\mathcal{L}_{qg} = -\sum_{i=1}^{N}\sum_{j=1}^{|q_i|} \log P\left(q_{i,j}|C_i, A_i, q_{i,<j-1}; \theta\right). \quad (2)$$

The fine-tuned KoBART shows a reasonable performance for question generation, but yet has three problems. One is that KoBART is not pre-trained with the sentences for question generation, another is that the learning objectives of KoBART is not directly related with question generation, and the other is that it often fails in grasping the structure of Korean interrogative sentences. Therefore, this paper solves these problems by post-training KoBART between the pre-training step and the fine-tuning step as shown in Figure 1.

After KoBART is pre-trained with Equation (1), it is post-trained with D_{qg} augmented by another data set D_{aug} using new objectives, *question infilling* and *Korean spacing*, for question generation. Then, the post-trained KoBART is fine-tuned again with Equation (2). The new objectives for post-training will be explained in the following section.

4 Post-Training Question Generator

The proposed post-training for question generation enhances the pre-trained KoBART in three ways. First, KoBART is allowed to experience the domain of question generation through post-training. Note that KoBART is not pre-trained with the sentences from question generation. Thus, KoBART is updated with D_{qg}, a data set for question generation. In order to make KoBART learn the domain of question generation effectively, a new objective, *question infilling* (QI), is proposed. Question infilling is equivalent to text infilling except that the MASK token can replace a word only at the question q, not in the context C. As a result, KoBART focuses more on a question than a context. This is achieved by a loss

$$\mathcal{L}_{kq} = -\sum_{(C_i,A_i,q_i)\in D_{qg}} \log P\left(q_i|C_i, q_i^c; \theta\right), \quad (3)$$

where q_i^c is a corrupted question of q.

When D_{qg} is small, the effect of question infilling is not definite. To increase the number of training instances, D_{qg} is augmented by another data set for question generation, D_{aug}. Then, Equation (3) is rewritten as

$$\mathcal{L}_{qi} = -\sum_{(C_i,A_i,q_i)\in D_{qg}\cup D_{aug}} \log P\left(q_i|C_i, q_i^c; \theta\right).$$

Even if KoBART is trained with Korean sentences, it often generates a grammatically wrong question. This is because KoBART does not capture the structure of questions perfectly. To solve this problem, KoBART is forced to learn how to space a word-concatenated sequence, since word spacing of questions helps KoBART understand the questions syntactically and semantically. In addition, word spacing is helpful for KoBART to find out which part of a context is related to a given question. This is achieved by introducing a new objective of *Korean spacing* formulated as

$$\mathcal{L}_{ks} = -\sum_{(C_i,A_i,q_i)\in D_{qg}\cup D_{aug}} \log P\left(q_i|C_i, q_i^{ks}; \theta\right),$$

where q_i^{ks} is a concatenated string of a question q_i.

To improve KoBART in all the three ways, KoBART is post-trained using both $\mathcal{L}_{qi}$ and $\mathcal{L}_{ks}$. That is, the final loss for KoBART post-training is

$$\mathcal{L}_{post} = \mathcal{L}_{qi} + \mathcal{L}_{ks}.$$

5 Experiments

5.1 Experimental Settings

Since there is no standard data set for Korean question generation, a new data set named as KorQuAD-QG is prepared from KorQuAD 1.0 (Lim et al., 2019) that contains 10,645 contexts. Each context can have multiple pairs of a question and an answer. As a result, KorQuAD has 66,181 pairs. Then, KorQuAD-QG is formulated as a set of triples of a context, a question, and an answer, where the context and the answer form an input for question generation and the question is an output. This KorQuAD-QG is used as D_{qg} to train the proposed question generator. Among 66,181 triples of

Model	BLEU-4	ROUGE-L	METEOR
Pre-trained KoBART	20.12	38.81	34.20
Post-trained KoBART	**21.05**	**40.07**	**34.82**

Table 1: Automatic evaluation results of the proposed question generator on KorQuAD-QG.

Model	Fluency	Relevancy
Pre-trained KoBART	4.55 ± 0.33	3.74 ± 0.12
Post-trained KoBART	$\mathbf{4.64 \pm 0.20}$	$\mathbf{3.93 \pm 0.14}$

Table 2: Human evaluations on one hundred questions sampled from KorQuAD-QG.

Model	BLEU-4	ROUGE-L	METEOR
Po.-T. KoBART	21.05	40.07	34.82
– QI	– 0.80	– 0.34	– 0.42
– DA	– 1.93	– 0.82	– 0.67
– KS	– 0.66	– 0.18	– 0.06
– (QI & DA)	– 0.94	– 1.16	– 0.75
– (KS & DA)	– 1.28	– 0.49	– 0.20

Table 3: The result of ablation study. "Po.-T. KoBART" is the post-trained KoBART, QI is *question infilling*, DA is *data augmentation*, and KS represents for *Korean spacing*.

KorQuAD-QG, 54,369 triples are used as a training set, 6,038 triples as a validation set, and the remaining 5,574 triples as a test set. The MRC data set from AI-Hub[2] with 243,425 triples is used for D_{aug}. The data sets are described in detail in appendix A.

KoBART is post-trained with the batch size of 16 and the sequence length of 512, while it is fine-tuned with the same batch size and sequence length. The beam search with the beam size of five is applied in decoding, and the AdamW (Loshchilov and Hutter, 2019) optimizer with the cosine warm-up scheduler is used for both post-training and fine-tuning where the initial learning rate is $3e-5$. All experiments below are done on a PC with one RTX-3090 GPU. All automatic evaluations are done with BLEU-4, ROUGE-L, and METEOR following Du et al. (2017).

5.2 Experimental Results

Table 1 summarizes the performance of the proposed question generator. The KoBART post-trained with the proposed objectives achieves 21.05 of BLEU-4, 40.07 of ROUGE-L, and 34.82 of METEOR, while the pre-trained KoBART shows just 20.12 of BLEU-4, 38.81 of ROUGE-L, and 34.20 of METEOR. That is, the post-trained KoBART outperforms the KoBART for all metrics. The difference between them is 0.93 BLEU-4, 1.26 ROUGE-L, and 0.62 METEOR, which proves the effectiveness of the proposed post-training. All these results are statistically significant (p-value < 0.05).

Human evaluation of the post-trained KoBART is given in Table 2. Three human evaluators compared the post-trained KoBART with the pre-trained KoBART for fluency and relevancy on 5-point scale with one hundred questions sampled from the test set of KorQuAD-QG. According to this table, the post-trained KoBART achieves 0.09 higher fluency and 0.19 higher relevancy than the pre-trained KoBART. Higher improvement in relevancy proves that the proposed post-training is effective in understanding interrogative sentences.

This paper has proposed three strategies of *question infilling* (QI), *data augmentation* (DA), and *Korean spacing* (KS) for post-training KoBART. In order to see the effectiveness of each strategy, an ablation study is performed and the result is shown in Table 3. '– QI' implies that KoBART is post-trained without $\mathcal{L}_{qi}$ and '– KS' means that it is post-trained without $\mathcal{L}_{ks}$. In both cases, DA is applied to post-training. '– DA' implies that D_{aug} is not used for post-training.

All 'QI', 'DA', and 'KS' are effective in improving KoBART, but 'DA' is proven to be most effective since the KoBART post-trained without 'DA' results in the largest performance degrade in all metrics. Transformer-based language models are sensitive to a data size. Thus, it requires a number of training instances to adapt itself to question generation. This is why 'DA' is the most important component for performance improvement by post-training of KoBART.

6 Conclusions

This paper has proposed a novel post-training of the pre-trained KoBART for Korean question generation. The proposed post-training enhances the pre-trained KoBART in three ways. First, by *question infilling*, the post-trained KoBART could not only be adapted to question generation, but also focus on the context area which is related to a question. Second, by learning *Korean spacing*, the post-trained

[2]https://aihub.or.kr

KoBART understands the Korean interrogative sentences semantically and semantically better than the pre-trained KoBART. Lastly, since transformer-based language models are sensitive to the number of training instances, the data set for question generation is augmented with additional MRC data. This data augmentation is empirically proven to be most effective in enhancing KoBART for question generation. In addition, since there is no standard data set for Korean question generation, this paper proposed a new data set of KorQuAD-QG for the task.

Acknowledgements

This work was partly supported by Institute of Information & communications Technology Planning & Evaluation (IITP) grant funded by the Korea government(MSIT) (No. 2022-0-00951, Development of Uncertainty-Aware Agents Learning by Asking Questions, 50%) and (No.RS-2022-00155911, Artificial Intelligence Convergence Innovation Human Resources Development (Kyung Hee University), 50%)

References

Ying-Hong Chan and Yao-Chung Fan. 2019. A Recurrent BERT-based Model for Question Generation. In *Proceedings of the 2nd Workshop on Machine Reading for Question Answering*, pages 154–162.

Li Dong, Nan Yang, Wenhui Wang, Furu Wei, Xiaodong Liu, Yu Wang, Jianfeng Gao, Ming Zhou, and Hsiao-Wuen Hon. 2019. Unified Language Model Pre-training for Natural Language Understanding and Generation. In *Advances in Neural Information Processing Systems 32*, pages 13042–13054.

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*, pages 1342–1352.

Nan Duan, Duyu Tang, Peng Chen, and Ming Zhou. 2017. Question Generation for Question Answering. In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*, pages 866–874.

Bilal Ghanem, Lauren Lutz Coleman, Julia Rivard Dexter, Spencer von der Ohe, and Alona Fyshe. 2022. Question Generation for Reading Comprehension Assessment by Modeling How and What to Ask. In *Findings of the Association for Computational Linguistics*, pages 2131–2146.

Jing Gu, Mostafa Mirshekari, Zhou Yu, and Aaron Sisto. 2021. ChainCQG: Flow-Aware Conversational Question Generation. In *Proceedings of the 16th Conference of the European Chapter of the Association for Computational Linguistics*, pages 2061–2070.

Suchin Gururangan, Ana Marasović, Swabha Swayamdipta, Kyle Lo, Iz Beltagy, Doug Downey, and Noah A. Smith. 2020. Don't Stop Pretraining: Adapt Language Models to Domains and Tasks. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 8342–8360.

Janghoon Han, Taesuk Hong, Byoungjae Kim, Youngjoong Ko, and Jungyun Seo. 2021. Fine-grained Post-training for Improving Retrieval-based Dialogue Systems. In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics*, pages 1549–1558.

Philippe Laban, John Canny, and Marti A. Hearst. 2020. What's The Latest? A Question-driven News Chatbot. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 380–387.

Guillaume Le Berre, Christophe Cerisara, Philippe Langlais, and Guy Lapalme. 2022. Unsupervised Multiple-Choice Question Generation for Out-of-domain Q&A Fine-tuning. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics*, pages 732–738.

Mike Lewis, Yinhan Liu, Naman Goyal, Marjan Ghazvininejad, Abdelrahman Mohamed, Omer Levy, Veselin Stoyanov, and Luke Zettlemoyer. 2020. BART: Denoising Sequence-to-Sequence Pre-training for Natural Language Generation, Translation, and Comprehension. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 7871–7880.

Seungyoung Lim, Myungji Kim, and Jooyoul Lee. 2019. KorQuAD1.0: Korean QA Dataset for Machine Reading Comprehension. arXiv:21909.07005.

Ilya Loshchilov and Frank Hutter. 2019. Decoupled Weight Decay Regularization. In *Proceedings of the 7th International Conference on Learning Representations*.

Baolin Peng, Chunyuan Li, Jinchao Li, Shahin Shayandeh, Lars Liden, and Jianfeng Gao. 2021. Soloist: Building task bots at scale with transfer learning and machine teaching. *Transactions of the Association for Computational Linguistics*, 9:807–824.

Yu Sun, Shuohuan Wang, Shikun Feng, Siyu Ding, Chao Pang, Junyuan Shang, Jiaxiang Liu, Xuyi Chen, Yanbin Zhao, Yuxiang Lu, Weixin Liu, Zhihua Wu, Weibao Gong, Jianzhong Liang, Zhizhou Shang, Peng Sun, Wei Liu, Xuan Ouyang, Dianhai Yu, Hao Tian, Hua Wu, and Haifeng Wang. 2021.

ERNIE 3.0: Large-scale Knowledge Enhanced Pre-training for Language Understanding and Generation. arXiv:2107.02137.

Ruize Wang, Duyu Tang, Nan Duan, Zhongyu Wei, Xuanjing Huang, Jianshu Ji, Guihong Cao, Daxin Jiang, and Ming Zhou. 2021. K-Adapter: Infusing Knowledge into Pre-Trained Models with Adapters. In *Findings of the Association for Computational Linguistics*, pages 1405–1418.

Taesun Whang, Dongyub Lee, Chanhee Lee, Kisu Yang, Dongsuk Oh, and Heuiseok Lim. 2020. An Effective Domain Adaptive Post-Training Method for BERT in Response Selection. In *Proceedings of the Annual Conference of the International Speech Communication Association*, pages 1585–1589.

Dongling Xiao, Han Zhang, Yukun Li, Yu Sun, Hao Tian, Hua Wu, and Haifeng Wang. 2020. ERNIE-GEN: An Enhanced Multi-Flow Pre-training and Fine-tuning Framework for Natural Language Generation. In *Proceedings of the 29th International Joint Conference on Artificial Intelligence*, pages 3997–4003.

Hu Xu, Bing Liu, Lei Shu, and Philip Yu. 2019. BERT Post-Training for Review Reading Comprehension and Aspect-based Sentiment Analysis. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics*, pages 2324–2335.

A Appendices

MRC data set from AI-Hub (MRC-AI-Hub) was used to support KorQuAD-QG data set during post-training. Even if both data sets are generated from question-answering data sets and share the same format, they have different characteristics.

- KorQuAD is constructed from Wikepeida pages, while AI-Hub is done from Korean news articles. Thus, the context of KorQuAD is usually much longer than that of AI-Hub. (see Table 4.)
- The number of questions in AI-Hub is much larger than that of KorQuAD. (refer to Section 5.1.) This is due to two reasons. One is that the number of news articles is much larger than that of Wikipedia pages. The other is that AI-Hub is prepared with more volunteers, since this data set was led by Korean government.
- While KorQuAD is constructed following the guide manual of SQuAD, AI-Hub is not. As a result, many questions of AI-Hub can be simply inferred from just a single sentence. For instance, in Table 4, the answer of 'World Health Organization' can be inferred from the clause "*The World Health Organization warns a possible massive epidemic and medical officials in the eastern region said that diarrhea, hepatitis and typhus are already spreading rapidly.*".

<table>
<tr><th colspan="2">KorQuAD-QG</th></tr>
<tr><td>Korean</td><td>Context: 양측 모두 경기의 어떤 시점에서든지 기권을 선언할 수 있다. 기권했을 경우 경기는 바로 종료되며, 기권한 사람의 패배가 된다. 일반적으로 자신이 이길 수 없거나 이길 가능성이 매우 희박하다고 생각할 때 기권을 선언한다. 기권을 선언할 때는 기권한다고 말을 하거나 기보에 기권한 것을 적으면 된다. 기보에 적을 때는 (1)영어로 기권한다는 뜻의 "resigns"라고 적는다, (2) 경기 결과에 동그라미를 친다, (3) 흑이 기권했을 경우 "1-0", 백이 기권했을 경우 "0-1"이라고 적는다. 자신의 킹을 넘어뜨리는 것도 기권을 뜻하지만 자주 사용되지 않는 방법이다. 심판을 부르기 위해서 양측 시계를 멈추기도 하기 때문에 양측 선수의 시계를 멈추는 것은 기권의 뜻이 아니다. 악수를 권유하는 것은 기권과 함께 많이 이루어지는데 이는 기권의 뜻라고 할 수 없다. 상대 선수가 악수의 의미를 기권이 아닌 무승부 요청으로 받아들일 수도 있기 때문이다.
Answer: resigns
Question: 기권을 선언할 때 영어로 기권한다는 뜻의 단어는?</td></tr>
<tr><td>English</td><td>Context: Either player may resign at any time, conceding the game to the opponent. If a player resigns, the game ends immediately and the player who resigns loses. In general, a player resigns when the player thinks the player cannot win or has a very slim chance of winning. A player may resign by saying it verbally or by indicating it on the score sheet in any of three ways: (1) by writing "resigns", (2) by circling the result of the game, or (3) by writing "1–0" if Black resigns or "0–1" if White resigns. Tipping over the king also indicates resignation, but it should be distinguished from accidentally knocking the king over. Stopping both clocks is not an indication of resigning, since clocks can be stopped to call the arbiter. An offer of a handshake is sometimes used, but it could be mistaken for a draw offer.
Answer: resigns
Question: What is the English word that a player writes on the chess notation for his resignation?</td></tr>
<tr><th colspan="2">MRC-AI-HUB</th></tr>
<tr><td>Korean</td><td>Context: 전염병 또한 심각한 문제다. 세계보건기구가 대규모 전염병 발생 가능성을 경고한 가운데, 동부 지역의 의료 관계자들은 이미 설사병, 간염, 티푸스 등의 돌림병이 빠른 속도로 확산되고 있다고 말했다.
Answer: 세계보건기구
Question: 대규모 전염병 발생 가능성을 경고한 곳은?</td></tr>
<tr><td>English</td><td>Context: Infectious diseases are also a serious problem. The World Health Organization warns a possible massive epidemic and medical officials in the eastern region said that diarrhea, hepatitis and typhus are already spreading rapidly.
Answer: World Health Organization
Question: Which organization has warned a possible massive epidemic?</td></tr>
</table>

Table 4: Examples of KorQuAD-QG and MRC-AI-Hub