# Semi-supervised Training Data Generation for Multilingual Question Answering

**Kyungjae Lee, Kyoungho Yoon, Sunghyun Park, Seung-won Hwang**
Yonsei University, Department of Computer Science
Yonsei-ro 50, Seoul, South Korea
{lkj0509, e05087, pshy410, seungwonh}@yonsei.ac.kr

**Abstract**

Recently, various datasets for question answering (QA) research have been released, such as SQuAD, Marco, WikiQA, MCTest, and SearchQA. However, such existing training resources for these task mostly support only English. In contrast, we study semi-automated creation of the Korean Question Answering Dataset (K-QuAD), by using automatically translated SQuAD and a QA system bootstrapped on a small QA pair set. As a naïve approach for other language, using only machine-translated SQuAD shows limited performance due to translation errors. We study why such approach fails and motivate needs to build seed resources to enable leveraging such resources. Specifically, we annotate seed QA pairs of small size (4K) for Korean language, and design how such seed can be combined with translated English resources. These approach, by combining two resources, leads to 71.50 F1 on Korean QA (comparable to 77.3 F1 on SQuAD).

**Keywords:** QA dataset, Machine Comprehension, Multilingual Resource

## 1. Introduction

Understanding text and answering questions about the text is an important yet challenging task for machines. To train machines for doing that, large-scale resources are necessary. For English, SQuAD (Rajpurkar et al., 2016) provides more than 100K pairs of questions and supporting passages: For example, for the question, such as "what causes precipitation to fall?", they locate the supporting "span", or the relevant answer found in the passage, such as "precipitation ... falls under gravity". As the advent of deep learning techniques turns algorithms data hungry, building resources of a significant size is a critical task, question answering (QA) systems, as demonstrated not only by SQuAD, but also by ImageNet for object recognition (Deng et al., ).

However, existing datasets for these tasks are mostly built on English. For other languages, building resources of large scale is labor intensive. Meanwhile, some approaches transfer English resources to resource-poor language for NLP tasks. As improvement of neural-based Machine Translation (MT) (Bahdanau et al., 2014) and availability for 100+ languages, some research takes advantage of MT systems for multilingual tasks such as sentiment analysis (Balahur and Turchi, 2014), query-document relevance (Ture and Boschee, 2014), and named entity recognition (Dandapat and Way, 2016). Given this opportunity, our research questions are as follow:

- **RQ1:** Can we build QA resource for other language, solely by automatically translating English resources?
- **RQ2:** When annotating QA resource manually for other language, can we save efforts by leveraging existing resources for English?
- **RQ3:** How can we combine the two to complement each other?

Regarding **RQ1**, we study **M**achine-**T**ranslated resource of a SQuAD, denoted as **MT**. Assuming perfect translation, **MT** should be sufficient to train a QA system of comparable performance to English QA trained on SQuAD. However, there are two challenges: (1) The position of answer span changes or is lost in translation. (2) Translation quality varies over QA pairs. Without overcoming these challenges, QA performance on **MT** is merely 52.49 in F1 (while 77.3 for SQuAD).

Regarding **RQ2**, the low performance on **MT** naturally motivates building language-specific (Korean) resource for QA. We construct and release such QA resource annotated by multiple human annotators, which we explain in Section 3. More interestingly, we show that, by selectively leveraging **MT**, manual effort for annotating Korean resources decreases drastically, from 100K+ pairs of SQuAD to mere 4K.

Finally, for **RQ3**, we train a QA system for Korean QA, combining both small-scale annotated data and large-scale translated resources we discussed above. As motivated from RQ1, there still remains the problem that the quality of the translation varies. To overcome this problem, we predict **translation certainty** for QA pairs, to use only high-quality QA pairs for training. With such selective training, we achieve 71.50 in F1 score.

## 2. Related Works

The availability of training resources has been driving QA research, ranging from early and small datasets such as WikiQA (Yang et al., 2015), to more recent and big datasets such as SQuAD (Rajpurkar et al., 2016), MS Marco (Nguyen et al., 2016), and SearchQA (Dunn et al., 2017) dataset. Our work complements these efforts by studying how to annotate small resources while selectively leveraging large resources developed for another language.

Most of existing competitive models for QA systems build on neural attention mechanism, to guide systems to focus on a targeted area in the passage. A baseline we adopt in this category is BiDAF model (Seo et al., 2016) which employs variant co-attention mechanism to match the question

Table 1: Statistics of dataset

| | Human dataset | Translated dataset | Total |
|---|---|---|---|
| # of passages | 1.4K | 19.8K | 21.2K |
| # of Q-A pairs | 4K | 77K | 81K |
| Avg length of question | 6.10 | 5.86 | - |
| Avg length of answer | 3.69 | 1.81 | - |

and passage mutually. Alternatively, R-Net adopts a gated attention-based recurrent network, with the gate modeling the importance of passage parts to the particular question, as such importance can differ for reading comprehension and question answering purposes respectively.

# 3. Model

## 3.1. Preliminary

As a basic QA model, we adopt Bi-Directional Attention Flow model (BiDAF) (Seo et al., 2016), with the highest accuracy for SQuAD among open-sourced implementations.[1] Two metrics are used to evaluate models: Exact Match (EM) and a softer metric, F1 score, which measures the weighted average of the precision and recall rate at word level. This model (by itself without ensemble) achieves an **EM score of 68.0 and an F1 score of 77.3**.

The inputs of this model are passage and question representation as both character- and word-level embeddings. They pass through bi-directional LSTM with attention to obtain a query-aware context representation. The output layer of the model is the probability distribution of the start and the end indices of the answer in the passage. The probabilities are calculated using the equation below:

$$P(y_1 = s) = softmax(w_1 * [G; M]) \quad (1)$$

$$P(y_2 = e) = softmax(w_2 * [G; LSTM(M)]) \quad (2)$$

where $s$ and $e$ indicate the start and the end position of answer span, G is the output of attention flow module according to passage, M is the output that is passed through two layers of LSTM according to the G, $w_1$ and $w_2$ are the weight of each output. In test, the model selects the answer span to maximize the product of two probabilities:

$$P(y_1 = s) * P(y_2 = e) \quad (3)$$

We will use this score to check translation quality of QA pairs in Section 3.3.

## 3.2. Dataset Construction

Our dataset consists of **MT** and **Seed**, representing machine-translated (large-scale) and manually annotated (small-scale) datasets respectively. Some statistics of the datasets are presented in Table 1.

### 3.2.1. MT Construction

For addressing **RQ1**, we study the challenge of translating answer spans in English into those in Korean translation. From observing Google Translate[2] of SQuAD QA pairs into Korean, we identify the following four cases:

[1] https://github.com/allenai/bi-att-flow

[2] https://translate.google.com

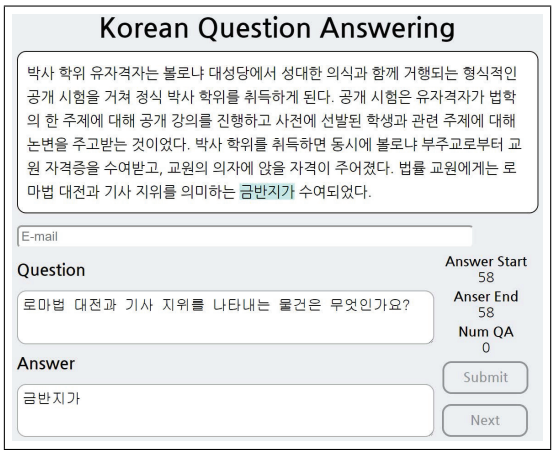


Figure 1: The web interface used to collect the Korean QA dataset. The labeling policy is based on SQuAD (Rajpurkar et al., 2016)

- Exact matching (35.5%): Terms in ESP (English answer spans) are translated into Korean terms which exactly match the Korean terms translating the matching English passage.
- Paraphrase matching (36.6%): Terms in KSP (Korean translated answer spans) are the paraphrase of terms used in Korean passage.
- Multiple spans (8.1%): Multiple ESPs are marked for a single sentence.
- Spans unpreserved: Google Translation cannot preserve answer spans in translation due to the language gap or translation inaccuracy.

To keep the first three cases (80.2% in total of SQuAD pairs) as MT resources, we adopt the following strategies. First case of exact matching can be trivially supported by string matching. For the second case, we mark answer spans in quotation, to serve the dual purposes of (1) giving a hint to translators to preserve the span boundary and (2) finding a matching paraphrase. For the third case, we preprocess QA pairs, to detach $k$ ESPs on the same sentence into $k$ pairs of one ESP and the sentence. Lastly, answer spans may be lost in translation, for which case, we do not use as training resources.

### 3.2.2. Seed Construction

The performance of QA, trained only on **MT**, is not effective with F1 score of 52.49, while that of English is 77.30. The main obstacle is poorly translated QA pairs, as shown in Figure 2(b), from which, neither machine nor human can find the right answer.

We thus build a small-scale **Seed** resources to serve dual purposes of (a) training a weak QA system and (b) predicting the translation quality of machine translated data. The advantage of **Seed** is near perfect precision, with the disadvantage of being labor intensive.

Our key contribution is to show that **Seed** does not have to be big and annotate such data ourselves. Unlike SQuAD

Passage : Napoleon maintained strict, efficient work habits, prioritizing what needed to be done. {…} Critics said he won many battles simply because of **luck**.

Question : According to critics, what was the reason Napoleon won many battles?

Passage : 나폴레옹은 엄격하고 효율적인 작업 습관을 유지하면서 해야 할 일에 우선 순위를 매겼습니다. {…} 비평가들은 단순히 **행운때문에** 많은 전투에서 승리했다고 말했다.

Question : 비평가들에 따르면, 나폴레옹이 많은 전투에서 승리 한 이유는 무엇입니까?

**Certainty: 0.994**

(a) example of excellent translation

Passage : During **World War II**, the island (Norfolk Island) became a key airbase and refuelling depot between Australia and New Zealand, and the Solomon Islands.

Question : During what major event did Norfolk Island become an important airbase and refuelling station?

Passage : **2 차 세계 대전기간** 동안 이 섬은 호주와 뉴질랜드, 뉴질랜드와 솔로몬 군도 사이의 주요 기지와 연료 보급 기지가 되었습니다.

Question : 노퍽 섬은 중요한 행사가 진행되는 동안 중요한 공군 기지 및 연료 보급소가 되었습니까?
Human re-translation : Did Norfolk Island become a major air base and fuel depot during important events?

**Certainty: 0.125**

(b) example of bad translation

Figure 2: Two examples of certainty of Korean QA pairs. Texts colored in red indicate answer span.

dataset containing 100K+ QA pairs, we show that the annotation of 4K is sufficient and release our annotation for future research[3].

To obtain seed QA pairs with comparable quality to SQuAD, we deploy a UI as shown in Figure 1, inspired by SQuAD conventions. We sample the 100 articles of Korean Wikipedia, to extract paragraphs of significant length without images. The result was 1464 paragraphs for the 100 articles covering a wide range of topics.

The UI in Figure 1 was deployed to human annotators, to read paragraph, enter questions, then highlight the spans including answers. Verbatim copying of Wikipedia text was discouraged, by disabling copy and pasting.

### 3.3. Translated Data Refinement

With **Seed**, we now study how to complement each other, rather than simply merge the two datasets. If mistranslated QA pairs are added to training set, the performance of the model may be worse, therefore we selectively denoise **MT** data as the quality of translation.

For such selection, we quantify **translation certainty**, as shown in Figure 2, high for correct translation (Figure 2(a)) and low for incorrect translation (Figure 2(b)). We use the weak QA system, or BiDAF built on **Seed** for such prediction. A naive solution is using Equation (3): Based on the starting and end positions $s$ and $e$, we use the equation to compare with the probabilistic certainty of the span predicted by the model, or BiDAF on the translated pair. This score will be high if our weak QA finds the right answer, which might be near impossible without reasonable translation quality. In other words, selecting QA pairs with high score would ensure high translation quality of the data.

However, this computation is rather strict, disallowing a minor disagreement in span boundaries, of including one more (or less) word before and after the span. We thus extend the equation to tolerate a minor error of one word, in both the start and end, or $s$ and $e$, as below:

$$\Big(P(y_1 = s-1) + P(y_1 = s) + P(y_1 = s+1)\Big) * \Big(P(y_2 = e-1) + P(y_2 = e) + P(y_2 = e+1)\Big) \quad (4)$$

Figure 2 shows the example scores this model computed: For example, in Figure 2(a) with high certainty, the passage and question were well translated into Korean, and a human can easily deduce correct answer through only this information. While, in Figure 2(b), the intention of the question was lost in translation, and the answer cannot be inferred from these question. This QA pair will not be used in training, guided by the low certainty score computed from the model.

## 4. Experiment

### 4.1. Experiment Setting

For test, we partitioned the annotated QA pairs randomly into a training set (2K) and a test set (2K), where the two sets do not share the same passages and articles. The implementation details used for this task are based on that of BiDAF model (Seo et al., 2016). We set a mini-batch size of 60 for 10 epochs on GPU Titan X, and a dropout rate of 0.5. Other hyper-parameters are the same with BiDAF model. For Korean-specific implementation of tokenizer and embedding, we adopt the state-of-the-arts KoNLPY[4] and skip-gram model (Mikolov et al., 2013) trained on Korean Wikipedia corpus and QA dataset.

### 4.2. Models

We compare our model with the following baselines:

- **Seed**: BiDAF using only manually annotated seed resources of a small scale.
- **MT**: BiDAF using only machine translated resources of a large scale.
- **Seed+Rand**: Hybridization of Seed and MT, by running BiDAF on Seed and $x\%$ of **Randomly** selected QA pairs from MT. For example, S+MT(25%) is the results of running BiDAF on Seed and 25% randomly selected QA pairs.

[3] https://e05087.github.io/

[4] http://konlpy-ko.readthedocs.io/

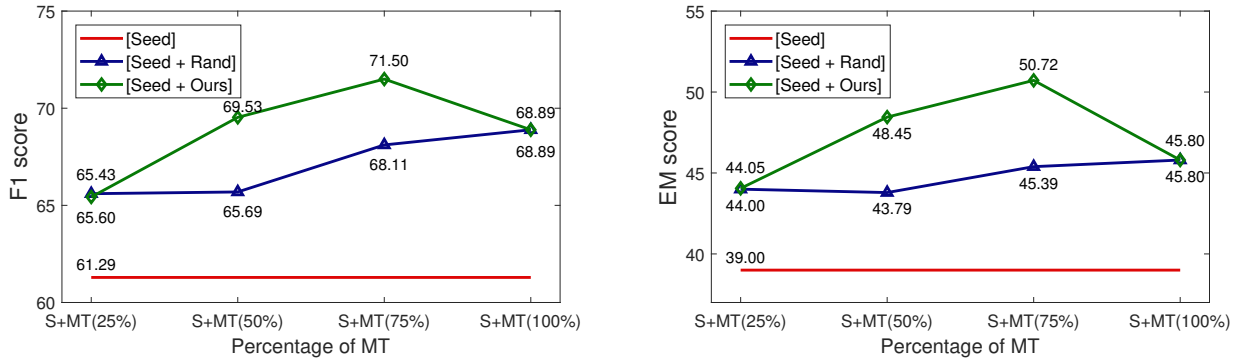


Figure 3: From left to right, the result of (a) F1 score and (b) EM score

Table 2: The results when using only Seed/MT, and ours which is the highest performance in our results.

| Metric | F1 score | Exact Match |
|---|---|---|
| Seed | 61.29 | 39.00 |
| MT | 52.49 | 10.13 |
| Ours | 71.50 | 50.72 |

Our proposed model prioritizes high-quality QA pairs from MT, and thus enables a **prioritized** selection of **MT** QA pairs. In this model, we use the techniques discussed in Section 3.3 to decide $x\%$ of the highest translation quality.

### 4.3. Result and Discussion

The results are reported in Table 2. In terms of both F1 and Exact Match (EM) scores, using only **Seed** or **MT** resources show poor performances, such as 61.29, 52.49 in F1 and 39.00, 10.13 in EM, respectively. In such case, the result on **Seed** is higher than that on **MT**, even though **Seed** is a small size compared to **MT**. This supports the need to build language-specific resource.

Meanwhile, as shown in Figure 3, using both **Seed** and **MT** set improves accuracy. As we add a random sampling of **MT** 25% set per each step, its performance also increases, and peaks at 68.89 in F1 and 45.80 in EM when $x = 100\%$. In the random sampling, the performances of F1 and EM tend to increase as the size of training data. Compared to the random sampling, our model, by refining **MT**, peaks at 71.50 in F1 and 50.72 in EM when $x = 75\%$. The peak value is higher than the result value at $x = 100\%$. This means that excluding refined 25% supports the improvement of the QA model. In all $x\%$ except at 25%, prioritized selection outperforms random selection. Although we use the same dataset and model, the prioritization of our approach enables a significant improvement, of F1 2.61 points increase. Therefore, our model is successful in effectively using small seed and large translation data to improve the performance of the QA model.

## 5. Conclusion

In this work, we study the feasibility of using translated resources for training QA systems. Inspired by our observations of challenges in using such translated resources for the task, we then build and release a 4K seed QA training resources for Korean language. We then study how we can combine such seed resources with the selective translated resources, for which we propose a model quantifying the translation certainty for the selective use of high quality resources. Lastly, we study the performance of QA systems on this combination of translated and seed resources. This release of seed resource and the proposed method of combining seed with large-scale resources available for another language is useful for follow-up research for providing QA services on many languages.

## 6. Acknowledgment

This research was supported by the MSIT(Ministry of Science and ICT), Korea, under the ITRC(Information Technology Research Center) support program(IITP-2017-2016-0-00464-002) supervised by the IITP(Institute for Information communications Technology Promotion)

## 7. References

Bahdanau, D., Cho, K., and Bengio, Y. (2014). Neural machine translation by jointly learning to align and translate. *arXiv preprint arXiv:1409.0473*.

Balahur, A. and Turchi, M. (2014). Comparative experiments using supervised learning and machine translation for multilingual sentiment analysis. *Computer Speech & Language*, 28(1):56–75.

Dandapat, S. and Way, A. (2016). Improved named entity recognition using machine translation-based cross-lingual information. *Computación y Sistemas*.

Deng, J., Dong, W., Socher, R., Li, L.-J., Li, K., and Fei-Fei, L. (). Imagenet: A large-scale hierarchical image database. In *CVPR 2009. IEEE Conference on*. IEEE.

Dunn, M., Sagun, L., Higgins, M., Guney, U., Cirik, V., and Cho, K. (2017). Searchqa: A new q&a dataset augmented with context from a search engine. *arXiv preprint arXiv:1704.05179*.

Mikolov, T., Sutskever, I., Chen, K., Corrado, G. S., and Dean, J. (2013). Distributed representations of words and phrases and their compositionality. In *NIPS*.

Nguyen, T., Rosenberg, M., Song, X., Gao, J., Tiwary, S., Majumder, R., and Deng, L. (2016). Ms marco: A human generated machine reading comprehension dataset. *arXiv preprint arXiv:1611.09268*.

Rajpurkar, P., Zhang, J., Lopyrev, K., and Liang, P. (2016). Squad: 100,000+ questions for machine comprehension of text. *arXiv preprint arXiv:1606.05250*.

Seo, M., Kembhavi, A., Farhadi, A., and Hajishirzi, H. (2016). Bidirectional attention flow for machine comprehension. *arXiv preprint arXiv:1611.01603*.

Ture, F. and Boschee, E. (2014). Learning to translate: a query-specific combination approach for cross-lingual information retrieval. In *EMNLP*, pages 589–599.

Yang, Y., Yih, W.-t., and Meek, C. (2015). Wikiqa: A challenge dataset for open-domain question answering. In *EMNLP*, pages 2013–2018.