QUAK: A Synthetic Quality Estimation Dataset for Korean-English Neural Machine Translation

Sugyeong Eo[1], **Chanjun Park**[1,2], **Hyeonseok Moon**[1], **Jaehyung Seo**[1], **Gyeongmin Kim**[1], **Jungseob Lee**[1], **Heuiseok Lim**[1]*
[1]Korea University, [2]Upstage
{djtnrud,bcj1210,glee889,seojae777,totoro4007,omanma1928,limhseok}@korea.ac.kr
chanjun.park@upstage.ai

Abstract

With the recent advance in neural machine translation demonstrating its importance, research on quality estimation (QE) has been steadily progressing. QE aims to automatically predict the quality of machine translation (MT) output without reference sentences. Despite its high utility in the real world, there remain several limitations concerning manual QE data creation: inevitably incurred non-trivial costs due to the need for translation experts, and issues with data scaling and language expansion. To tackle these limitations, we present QUAK, a Korean-English synthetic QE dataset generated in a fully automatic manner. This consists of three sub-QUAK datasets QUAK-M, QUAK-P, and QUAK-H, produced through three strategies that are relatively free from language constraints. Since each strategy requires no human effort, which facilitates scalability, we scale our data up to 1.58M for QUAK-P, H and 6.58M for QUAK-M. As an experiment, we quantitatively analyze word-level QE results in various ways while performing statistical analysis. Moreover, we show that datasets scaled in an efficient way also contribute to performance improvements by observing meaningful performance gains in QUAK-M, P when adding data up to 1.58M.

1 Introduction

Quality estimation (QE) is the task of predicting the translation quality as a continuous value or discrete tags by referring to a source sentence and its machine translation (MT) output (Blatz et al., 2004; Specia et al., 2009, 2013). Since quality annotations on MT output are applied in various ways according to the granularity levels (word, sentence, document, etc.), QE research has been constantly developing in recent years (Kim et al., 2017; Fomicheva et al., 2020a; Alva-Manchego et al., 2021; Ding et al., 2021b).

MT output	Given that the Chinese authorities do not deny it , it is highly likely .
pseudo-PE	Given that the Chinese authorities do not deny it , chances are high .
MT output tags	OK OK OK OK OK OK OK OK OK OK OK OK OK OK OK OK OK OK OK OK OK BAD OK BAD OK BAD OK BAD OK OK OK
Source	중국 당국이 부인하지 않는 것으로 볼 때 가능성이 높다 .
Source tags	OK OK OK OK OK OK OK BAD BAD OK
Alignments	0-3 1-4 2-7 3-5 3-6 4-8 5-8 6-0 7-13 8-11 8-12 9-14
Edits	(1) Insertion (‘ ’→ it) (2) Substitution (chances→is) (3) Substitution (are→highly) (4) Substitution (high→likely)

Table 1: An example of QUAK dataset. For the correct translation, one insertion and three substitutions are required for the MT output. Although not included in this example, if there is a missing word, a BAD tag is attached to the location of the corresponding gap token. We indicate the alignment information (Alignments) in the form of {source index}-{aligned MT output index}.

Owing to this importance, datasets for training QE systems are being released continuously. However, we highlight three limitations for the existing QE dataset. (1) First, non-trivial human labor and time cost are required when constructing data. Source sentences, MT output, and quality annotations are dataset prerequisites for QE learning, among which translation experts proficient in a language pair are essential in the labeling process. Employing experts is far more difficult especially in low-resource languages.

(2) As an extension of the first limitation, manual QE datasets are restricted in size regardless of the data resource. The meticulous work of creating human post-edited sentences with minimal modifications slows down the construction time, which makes it difficult to scale. Most released QE datasets, including those from the Conference on Machine Translation (WMT) are composed of data less than 10K in size (Fujita and Sumita, 2017; Fomicheva et al., 2020b). This is a much rarer amount compared with the large volumes of data used by studies on GPT 3 (Brown et al., 2020) in terms of data-hungry NLP.

(3) Available QE language pairs are limited. Although the released WMT QE dataset considers

* Corresponding Author

Proceedings of the 29th International Conference on Computational Linguistics, pages 5181–5190
October 12–17, 2022.

high, medium, and low resources (Fomicheva et al., 2020c), it still covers only a much smaller number of language pairs compared with parallel corpora. Since data construction in opposite directions for a language pair requires an entirely different human post-edited sentence, numerous language pairs and directions are yet to be utilized.

To mitigate the above limitations, we introduce QUAK [1], a large-scale Korean-English synthetic QE dataset. This is built as an automated process by taking the Eo et al. (2021a) approach and aims to train word-level QE. Namely, the data generation process does not demand human post-editing, allowing data to be built at scale than manual methods. In addition, language extension is relatively free as it is language-agnostic within a language pair in which Google translation is possible and a corresponding corpus exists. Therefore, we adopt Korean-English, one of the morphologically rich languages rarely addressed in the QE field.

For constructing QUAK, A monolingual or parallel corpus and an MT model are required. QUAK is divided into three sub-QUAK datasets according to data sources: (1) QUAK-Monolingual (QUAK-M) leveraging a monolingual corpus of the target language, (2) QUAK-Parallel (QUAK-P) leveraging a parallel corpus, and (3) QUAK-Hybrid (QUAK-H) jointly leveraging monolingual and parallel corpus. The final QUAK training data size in QUAK-P and QUAK-H is 1.58M and 6.58M in QUAK-M, which is about 225 times and 940 times larger than the 7K size of the WMT official dataset (Fomicheva et al., 2020b).

Considering that QUAK is synthetic data, we scrutinize the dataset with statistics and a quantitative analysis to provide reliability and quality assurance. In the quantitative analysis, in particular, we first compare the word-level QE model fine-tuning performance based on multiple multilingual pre-trained language models (mPLMs) using only 100K pieces from each sub-QUAK. Thereafter, we use the best performing model to incrementally scale the data size and track performance fluctuations.

As a result of the experiment, the XLM-RoBERTa (XLM-R) (Conneau et al., 2019) large model is the most competitive, showing a difference of up to 0.12 MCC compared to other mPLMs such as multilingual BART (mBART) (Liu et al., 2020), XLM (Lample and Conneau, 2019). Furthermore, scaling the data to 1.58M tends to improve overall model performance. Based on the MT output-side, QUAK-M obtains performance gain of maximum 0.042, QUAK-P 0.037, and QUAK-H 0.029 MCC. Our contributions are as follows:

- To minimize the exorbitant human-demand and time cost of QE, we construct and release the QUAK dataset in a fully automatic manner, exploiting three efficient data generation strategies.
- To address the size limitation, we expand the synthetic data to a large-scale. We scale QUAK up to 940 times compared with the WMT official dataset.
- As the language pair for QUAK, we choose Korean-English, a low-resource language pair that has never been released before. Language coverages can be extended if only the translation model and its corpus are satisfied in the generation process.
- We analyze the QE fine-tuning performance according to various mPLMs, and analyze the results of progressively expanding the data for the best performing QE model.

2 Related Work

As QE research has increasingly been introduced recently, human-labeled QE datasets are also being released (Specia et al., 2010; Fujita and Sumita, 2017; Fomicheva et al., 2020c,b). However, data construction processes have several limitations in terms of time cost, data size, and available language pairs.

Many studies have been conducted continuously to handle these limitations. To name a few, Tuan et al. (2021) propose a synthetic data construction method that utilizes a parallel corpus to alleviate the cost of human labor and time cost. In such study, translation errors committed through a language model or an NMT system are injected to parallel sentences. A similar method of generating synthetic data through parallel corpus has also been leveraged in automatic post-editing research (Negri et al., 2018).

To address data size restrictions caused by time cost in human annotations, attempts have been made using data augmentation (Lee, 2020; Wang

[1] Our QUAK dataset is publicly available at https://bit.ly/3dqe2KE.

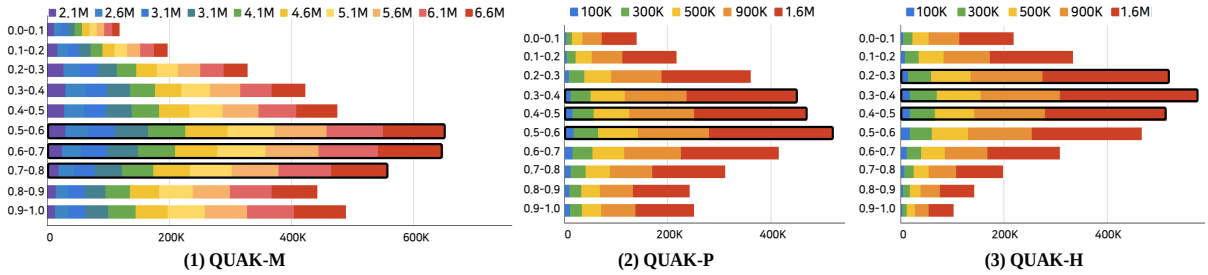

Figure 1: Distribution of the data size according to the TER range for each sub-QUAK. We present the top three scopes with the largest amount in bold lines.

et al., 2020; Gajbhiye et al., 2021; Ding et al., 2021a) and unsupervised learning (Etchegoyhen et al., 2018). In the study of Fomicheva et al. (2020c), unsupervised quality indicators based on uncertainty quantification are exploited to train the QE model.

To tackle constraints about available language pairs, cross-lingual zero-shot QE approaches are constantly studied (Sun et al., 2020; Eo et al., 2021b). Following this trend, WMT21 includes zero-shot to their main interests, evaluating the competence of a QE model on unseen languages (Specia et al., 2021). This study addresses all of the respective factors and presents QUAK.

3 QUAK

QUAK is a QE dataset for evaluating the quality of Korean-English MT output, and includes three sub-QUAKs according to the data selection. Each sub-QUAK comprises *(1) a source sentence, (2) its MT output, and (3) OK/BAD quality annotation*. Source sentence and its MT output are utilized as model input, allowing the model to classify the quality of these sentences into OK/BAD tags on a token basis. Quality annotations separately exists for source sentence and MT output. When the model predicts the translation quality, these are used as a ground-truth for evaluation. With primary consideration of efficiency and effectiveness, we construct data in a fully-automatic manner, exploited method by Eo et al. (2021a). We select existing monolingual or parallel corpora for our data sources (detailed in Section 3.1) and use them to each data production process (detailed in Section 3.2).

3.1 Dataset Sources

QUAK is divided into three sub-QUAKs. The raw dataset requirements to build each sub-QUAK are as follows: QUAK-M requires a target language monolingual corpus, QUAK-P requires a parallel corpus, QUAK-H requires both corpora.

For a monolingual corpus, we adopt English Wikipedia, which consists of documents on a wide range of topics. We use it to handle the various translation errors that the MT model may commit to the diverse entities and expressions in Wikipedia. We randomly extract 5M sentences to generate QUAK-M. For a parallel corpus, we leverage AI hub parallel corpus released by Korea National Information Society Agency[2]. AI hub corpus also covers various fields such as news, journals, law, and culture, and is produced with high quality through human inspection. The parallel corpus contains 1,602,002 pairs.

For fair comparison with other sub-QUAKs, in the case of QUAK-M, we combine both AI hub and Wikipedia source. Namely, we configure 1.58M of QUAK-M using the target-side text of the Ai hub and the remaining 5M using Wikipedia. The validation and test set is configured by randomly selecting 12K pieces of AI hub data.

3.2 Dataset Construction Process

QUAK-M For QUAK-M, we utilize a target language monolingual corpus. With the text, we first conduct a round-trip translation. We translate the English corpus into Korean sentences, where we denote them as pseudo-source sentences. We once again forward-translate the pseudo-source to generate the MT output.

When pseudo-source and its MT output have been created, quality annotations are tagged. Prior to label annotation, we pre-define target language monolingual corpus to be a flawless sentence. Based on this assumption, we consider this text as a pseudo-post-edited (pseudo-PE) sentences for which correction has been completed. By com-

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

paring the MT output and pseudo-PE in a token-wise fashion, we measure the minimum substitution/deletion/insertion errors needed based on edit distance. The OK/BAD tag indicating correct/wrong translation for each token in the MT output is further annotated. If the number of MT output tokens in a sentence is N, the number of OK/BAD tags for this is $2N + 1$ because gap tokens are attached to the front and back of each MT output token. If there are missing words, BAD tags are added to the position of the corresponding gap token, otherwise OK tags are labeled. Apart from tagging the MT outputs, source tags are also annotated according to the binary tags of the MT outputs based on word alignment information. The tag for the source sentence excludes the gap token labeling.

QUAK-P The parallel corpus is leveraged in the QUAK-P configuration. This has higher connectivity between the source and target sides and has an intact source sentence compared with the pseudo-source of QUAK-M. To obtain QUAK-P, we proceed a one-way translation from the source to target language. In this case, source-side difference from QUAK-M leads to various translation results for the MT output. Similar to the QUAK-M generation process, we consider the target-side text of the parallel corpus as a pseudo-PE. With source sentences, its MT output, and pseudo-PE, we label the quality of the translation results. After calculating the minimum edit operation between the MT output and the pseudo-PE, BAD tags are attached to the token where the modification occurred. For quality annotations on source sentences, the same tags are attached to the MT output index and the aligned source index.

QUAK-H In QUAK-H, we combine the above two previous sub-QUAKs to generate various translation results with a limited corpus. We compose the source sentence and MT output by selectively utilizing two approaches proposed in QUAK-P and QUAK-M, respectively. Namely, we use the source-side text and pseudo-PE text from QUAK-P, and the MT output-side text from QUAK-M. By dealing with two different MT outputs with the same source-side text in QUAK-P, we induce the QE model to learn by referring to various combinations of the source sentence and MT output. For the next step, we tag labels for quality annotation as mentioned above.

Final constructed QUAK dataset After three construction processes, we obtain a total of 1,578,002 training examples for QUAK-P and QUAK-H, and 6,578,002 training examples for QUAK-M. We present an example of QUAK in Table 1. MT output for a source sentence "중국 당국이 부인하지 않는 것으로 볼 때 가능성이 높다 ." is mistranslated into "`Given that the Chinese authorities do not deny it, it is highly likely .`". The MT output should be corrected into "`chances are high`". This should perform a four minimum correction, which will result in a four BAD tags of the entire MT output tag. In addition, based on the word alignment "가능성(7)–`likely`(13), 높다(8)–`is`(11), 높다(8)–`highly`(12)", the BAD tag index of the MT output is also reflected in the source-side index.

4 Experimental Setup

4.1 Experimental Design

In this section, we present the statistical and quantitative analysis done on the QUAK. In the statistical analysis, we measure the sentence length, token length, and average token length per sentence for each sub-QUAK. We also calculate the mean, median, standard deviation, and variance of the translation edit rate (TER) score. Regarding tags, we count the total number of OK and BAD tags.

During the quantitative analysis, we experiment three word-level QE fine-tuning to efficiently achieve high performance and analyze large-scale QUAK data. In the first experiment, we fine-tune multiple mPLMs with 100K pieces of QUAK-M, P, H to explore which model performs better for QUAK.

Thereafter, we inspect the impact on the amount of QUAK. We fine-tune the data for the previous best performing model, scaling each sub-QUAK exponentially from 100K to 1.58M. As mentioned earlier, one consideration is that QUAK-M (1.58M) consists of target-side text in a parallel corpus for proper comparison with the data generated by other strategies.

In the last experiment, we gradually increase the size of QUAK-M. Our result includes the corresponding performance while extending from the previous size of 1.58M to 6.58M in 500K increments.

Attributes	Google	Amazon	Microsoft	Systran
# of Source Sentences	12,000	12,000	12,000	12,000
# of MT Output	12,000	12,000	12,000	12,000
# of pseudo-PE	12,000	12,000	12,000	12,000
# of Source Tokens	199,413	199,413	199,413	199,413
# of MT Output Tokens	340,264	303,535	325,973	346,030
# of pseudo-PE Tokens	342,385	342,385	342,385	342,385
Average Token Per Source Sentence	16.62	16.62	16.62	16.62
Average Token Per MT Output	28.36	25.29	27.16	28.84
Average Token Per pseudo-PE	28.53	28.53	28.53	28.53
Mean TER	0.57	0.63	0.63	0.46
Median TER	0.57	0.64	0.64	0.44
STD TER	0.23	0.21	0.21	0.26
Variance TER	0.05	0.04	0.05	0.07
# Source OK tags	112,647	94,562	97,825	134,503
# Source BAD tags	86,766	104,851	101,588	64,910
# MT Output OK tags	510,085	429,775	465,441	560,221
# MT Output BAD tags	182,443	189,295	198,505	143,839

Table 2: Statistics for the four test sets. We denote the target-side text of corpus as pseudo-PE.

4.2 Experimental Settings

Models In all experiments, we exploit the MicroTransQuest (Ranasinghe et al., 2021) framework. While it only uses an XLM-R model, we utilize additional mPLMs: In the QE model training, we leverage XLM, XLM-R, and mBART.

From Huggingface (Wolf et al., 2019), we load five mPLMs that have learned both Korean and English: xlm-mlm-100-1280, xlm-roberta-base, xlm-roberta-large, facebook/mbart-large-cc25, and facebook/mbart-large-50.

Datasets For the data construction, the following tools are used in this study. As monolingual data we dump Wikipedia and use Wikiextractor[3] to extract plain text. We train a Korean-English and English-Korean MT model using the fairseq (Ott et al., 2019) package with SentencePiece subword tokenization (Kudo and Richardson, 2018) to translate sentences. We train the word alignment between the source and target text using the FastAlign (Dyer et al., 2013) toolkit and measure the edit distance using Tercom software (Snover et al., 2006). Tag annotation is executed using the Unbabel corpus builder[4]. We adopt Mosesdecoder (Koehn et al., 2007) for additional data preprocessing.

Evaluation For constructing the test sets, we utilize a publicly available external machine translator to ensure the reliability and objectivity of the QE results. Four representative commercialized machine translators are adopted, including Google[5], Amazon[6], Microsoft[7], and Systran[8]. Through these, the test sets are established in the same manner as the strategy used in QUAK-P. The test sets are based on 12K sentence pairs randomly extracted from the Ai hub parallel corpus without overlapping with the training and validation sets. Matthews correlation coefficient (MCC) (Chicco and Jurman, 2020) is used as a metric for evaluating QE model performance.

Table 2 provides the test set statistics. When analyzing the number of OK/BAD tags, the results vary depending on the translator even when the same source sentence is used. Systran differs the most compared with other test sets: there are 143,839 MT output BAD tags with an average TER difference of 0.17 with the highest value of 0.63.

5 Analysis and Results

5.1 Data statistics and analysis

We report the statistics for QUAK in Table 3. QUAK-M additionally uses the English Wikipedia corpus consisting of 5M samples, and this yields different data sizes in contrast to other sub-QUAKs. Comparing training set of QUAK-M with other sub-QUAKs, the most dominant part is the relation between the average token length and TER score. QUAK-P and QUAK-H show lower TER scores even though their average tokens per sentence are relatively higher. We interpret this result as a case where the translation works well even if the average sentence length is long. As shown in Figure 1, comparing data sizes by TER range is also consistent with this statistic. The data is mainly concentrated in the 0.5–0.8 range for QUAK-M, 0.3–0.6 for QUAK-P, and 0.2–0.5 for QUAK-H. From this, we speculate that Wikipedia may have more noise in the text itself than the Ai hub, and that a large number of errors are committed during the translation process.

Next, for QUAK-P and QUAK-H, the number of BAD tags of QUAK-P is greater than that of QUAK-H in the MT output. It is noteworthy that the MT output of QUAK-H is created based on a round-trip translation, which is identical to that of QUAK-M. These indicate that the pseudo-source generated by the target language text of Ai hub is adequately restored to the original sentence when

[3] https://github.com/attardi/wikiextractor

[4] https://github.com/Unbabel/word-level-qe-corpus-builder

[5] https://translate.google.co.kr/?hl=en

[6] https://aws.amazon.com/translate/

[7] https://www.microsoft.com/en-us/translator/

[8] https://translate.systran.net/

	Train			Valid		
Attributes	**QUAK-M**	**QUAK-P**	**QUAK-H**	**QUAK-M**	**QUAK-P**	**QUAK-H**
# of Source Sentences	6,578,002	1,578,002	1,578,002	12,000	12,000	12,000
# of MT Output	6,578,002	1,578,002	1,578,002	12,000	12,000	12,000
# of pseudo-PE	6,578,002	1,578,002	1,578,002	12,000	12,000	12,000
# of Source Tokens	92,848,776	25,149,673	25,149,673	209,894	199,624	199,624
# of MT Output Tokens	139,620,328	42,051,001	39,850,492	318,959	340,855	318,921
# of pseudo-PE Tokens	148,922,086	42,103,966	42,103,966	342,021	341,996	341,996
Average Token Per Source Sentence	14.12	15.94	15.94	17.50	16.64	16.64
Average Token Per MT Output	21.23	26.65	25.25	26.58	28.40	26.58
Average Token Per pseudo-PE	22.64	26.68	26.68	28.50	28.50	28.50
Mean TER	0.61	0.50	0.42	0.45	0.57	0.45
Median TER	0.63	0.50	0.40	0.44	0.56	0.44
STD TER	0.24	0.25	0.23	0.21	0.23	0.21
Variance TER	0.06	0.06	0.05	0.04	0.05	0.04
# Source OK tags	50,421,860	15,600,416	16,269,784	84,873	113,192	121,700
# Source BAD tags	42,426,916	9,549,257	8,879,889	82,343	86,432	77,924
# MT Output OK tags	204,721,896	65,852,185	65,033,087	339,304	511,542	506,437
# MT Output BAD tags	81,096,762	19,827,819	16,245,899	157,308	182,168	143,405

Table 3: Statistics for three sub-QUAK training and validation set

translated back, even if it is different from the correctly translated source-side of the parallel corpus.

5.2 Experimental Results

Performance Comparison by mPLMs We provide the fine-tuning results for mPLMs by selecting only 100K of the datasets in Table 4. From the experimental results, XLM-R-large model shows the best performance. Based on the MT output-side MCC (Target MCC) of the Google test set, XLM-R-large reports 0.366, 0.401, and 0.324 for QUAK-M,P,H, and outperforms the XLM-R-base model by 0.023, 0.024, and 0.004, respectively. For the source-side MCC (Source MCC), XLM-R-large also achieves 0.285, 0.331, and 0.271 for QUAK-M,P,H, which are the best competencies compared to other models.

In all test sets, except for Systran, the Target MCC and Source MCC of XLM-R-large performed the best in all sub-QUAK datasets. XLM-R-large differs from XLM-R-base in terms of the number of parameters; the former contains 550M, whereas the latter has 270M. Based on the Target MCC of the Amazon test set, XLM-R-large generally reports a higher performance than XLM-R-base, achieving 0.035, 0.032, and 0.022 higher values for QUAK-M, P, and H, respectively. These results demonstrate that the number of parameters in mPLMs poses a positive effect on the QE model learning.

Regarding mBART and mBART50, the latter outperforms the former in general. This implies the substantial impact of the number of pre-trained

	Dataset	XLM-R -base	XLM-R -large	mBART	mBART50	XLM
Google						
Target MCC	QUAK-M	0.343	**0.366**	0.340	0.343	0.296
	QUAK-P	0.377	**0.401**	0.376	0.382	0.339
	QUAK-H	0.320	**0.324**	0.306	0.314	0.292
Source MCC	QUAK-M	0.279	**0.285**	0.275	0.276	0.231
	QUAK-P	0.315	**0.331**	0.309	0.320	0.285
	QUAK-H	0.266	**0.271**	0.258	0.267	0.249
Amazon						
Target MCC	QUAK-M	0.389	**0.424**	0.388	0.385	0.328
	QUAK-P	0.408	**0.440**	0.405	0.410	0.362
	QUAK-H	0.362	**0.384**	0.264	0.359	0.322
Source MCC	QUAK-M	0.324	**0.342**	0.320	0.323	0.254
	QUAK-P	0.353	**0.377**	0.341	0.354	0.304
	QUAK-H	0.305	**0.323**	0.213	0.310	0.276
Microsoft						
Target MCC	QUAK-M	0.380	**0.415**	0.382	0.380	0.315
	QUAK-P	0.401	**0.433**	0.404	0.406	0.346
	QUAK-H	0.353	**0.372**	0.253	0.355	0.316
Source MCC	QUAK-M	0.307	**0.329**	0.303	0.307	0.244
	QUAK-P	0.338	**0.363**	0.327	0.338	0.287
	QUAK-H	0.290	**0.310**	0.193	0.299	0.271
Systran						
Target MCC	QUAK-M	0.261	**0.277**	0.255	0.253	0.206
	QUAK-P	0.298	**0.311**	0.289	0.296	0.261
	QUAK-H	**0.226**	0.217	0.122	0.221	0.196
Source MCC	QUAK-M	**0.224**	0.223	0.218	0.221	0.161
	QUAK-P	0.247	**0.250**	0.228	0.242	0.210
	QUAK-H	**0.179**	0.174	0.076	0.176	0.159

Table 4: Comparison of word-level Korean-English QE performance by mPLMs fine-tuned with each sub-QUAK dataset

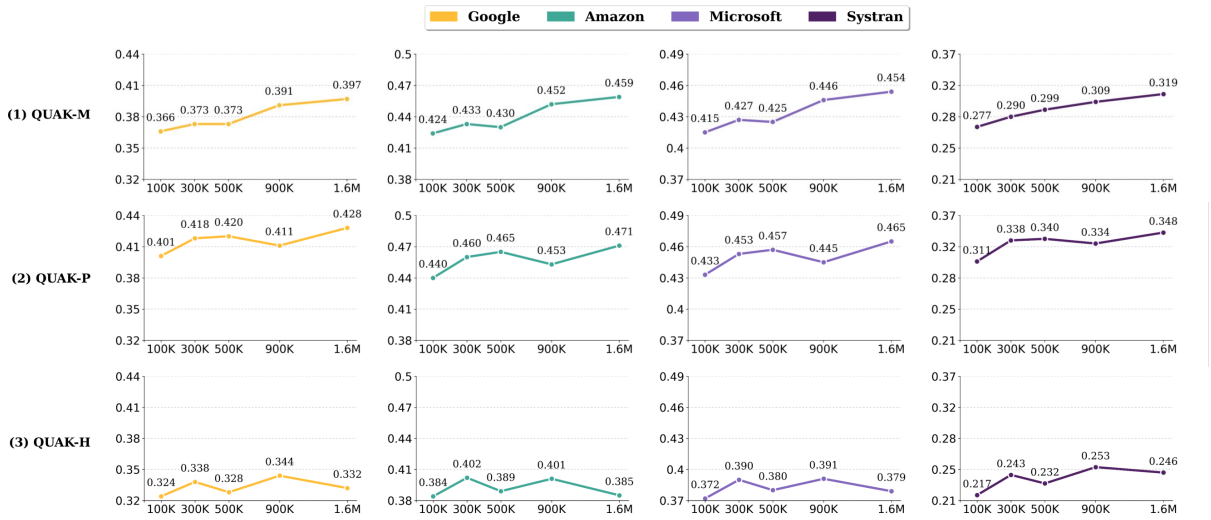

Figure 2: MCC variation of QUAK according to data scaling

	Google		Amazon		Microsoft		Systran	
Data Size	Target MCC	Source MCC	Target MCC	Source MCC	Target MCC	Source MCC	Target MCC	Source MCC
1.58M	0.397	0.324	0.459	0.386	0.454	0.376	0.319	0.273
2.08M	0.386	0.319	0.441	0.371	0.436	0.365	0.317	0.272
2.58M	0.384	0.316	0.442	0.372	0.436	0.364	0.308	0.264
3.08M	0.385	0.316	0.442	0.375	0.435	0.365	0.313	0.272
3.58M	0.381	0.314	0.443	0.374	0.437	0.365	0.311	0.269
4.08M	0.382	0.313	0.438	0.369	0.434	0.359	0.310	0.270
4.58M	0.376	0.310	0.433	0.368	0.426	0.357	0.307	0.267
5.08M	0.378	0.308	0.436	0.366	0.432	0.362	0.308	0.269
5.58M	0.377	0.319	0.439	0.374	0.432	0.364	0.307	0.274
6.08M	0.351	0.286	0.400	0.335	0.398	0.334	0.296	0.261
6.58M	0.387	0.325	0.451	0.379	0.441	0.369	0.313	0.274

Table 5: Performance variation of QUAK-M according to data scaling

languages. It is noteworthy that mBART50 is pre-trained for 50 languages, enabling more multilingual support than mBART, which is learned on 25 languages. As Korean is regarded as a relatively low-resource language and especially utilizes only 100K data, we infer that mBART50 has more influence on competence gain from high-resource languages than mBART.

Performance Comparison for Scaling The previously obtained results show that the XLM-R-large model is superior to all 100K sub-QUAK datasets. For the next experiment, we explore the performance fluctuation by constantly increasing the size of the QUAK dataset to XLM-R-large. Figure 2 illustrates the variation of the performance depending on the corpus size. The experimental results demonstrate that the performance variation tends to be similar for all test sets.

When we exponentially scale the data for the three sub-QUAKs, QUAK-M had a notable achievement. Furthermore, QUAK-P showed a steady increase, except for the case of 900K. We confirm that data scaling is one factor in increasing the performance of the QE model. However, in the case of QUAK-H, there is no clear trend in terms of data expansion. We argue that although the MT output is applied owing to various translations for source sentences, the weakened connectivity between two sentences might impede learning.

Performance of QUAK-M (6.58M) QUAK-M requires only monolingual corpus in the data building process. This allows data size expansion over other sub-QUAKs that utilize a parallel corpus. Exploiting these, we further extend the Wikipedia corpus by 5M, comprising a total of 6.58M. We gradually add data in 500K increments to check the performance fluctuation.

The experimental result is presented in Table 5. The target MCC performance on the Google test set with 1.58M is lower by -0.016 at 3.58M(+2M) and -0.020 at 5.58M(+4M). The performance of Amazon, Microsoft, and Systran also degraded by -0.016, -0.017, -0.008 at 3.58M and -0.02, -0.022, and -0.012 at 5.58M compared to 1.58M, respectively. We observe that the overall QE model performance has deteriorated as more data is added.

We interpret this result in terms of data. QUAK-H, P, and the test sets are extracted from the Ai hub dataset. As observed in the previous statistics (Table 3), this resulted in a difference in terms of average TER in QUAK-M, which also contains

	TER Range									
Data Size	**0.0-0.1**	**0.1-0.2**	**0.2-0.3**	**0.3-0.4**	**0.4-0.5**	**0.5-0.6**	**0.6-0.7**	**0.7-0.8**	**0.8-0.9**	**0.9-1.0**
QUAK-P										
100K	0.214	0.323	0.361	0.398	0.410	0.382	0.374	0.338	0.308	0.262
1.58M	0.209	0.348	0.386	0.420	0.438	0.413	0.395	0.377	0.325	0.279
Diff	-0.005	0.025	0.025	0.022	**0.028**	**0.031**	0.021	**0.039**	0.017	0.017
QUAK-H										
100K	0.206	0.312	0.335	0.378	0.375	0.337	0.312	0.263	0.237	0.172
1.58M	0.192	0.316	0.330	0.366	0.374	0.337	0.309	0.275	0.249	0.192
Diff	-0.014	0.004	-0.005	-0.012	-0.001	0.00	-0.003	**0.012**	**0.012**	**0.020**
QUAK-M (1.58M)										
100K	0.210	0.295	0.342	0.380	0.393	0.361	0.339	0.303	0.269	0.223
1.58M	0.215	0.329	0.361	0.398	0.413	0.381	0.366	0.340	0.312	0.265
Diff	0.005	0.034	0.019	0.018	0.020	0.020	0.027	**0.037**	**0.043**	**0.042**
QUAK-M (6.58M)										
100K	0.210	0.295	0.342	0.380	0.393	0.361	0.339	0.303	0.269	0.223
6.58M	0.184	0.308	0.352	0.400	0.399	0.377	0.364	0.339	0.298	0.261
Diff	-0.026	0.013	0.010	0.020	0.006	0.016	0.025	**0.036**	**0.029**	**0.038**

Table 6: Target MCC performance difference (Diff) by TER range for Google test set. When we add data, we underline the three cases with the worst performance, and bold the three cases with the most performance improvement.

Wikipedia. QUAK-M is mainly distributed in a range with a high TER score, while QUAK-H, QUAK-P, and the test set are included in relatively low scores. This indicates that the difference from the test set in terms of data distribution also affected the performance of QUAK-M.

Performance Comparison by TER Range In addition to the previous results, we divide the Google test set into units of 0.1 TER for more precise comparison. We then verify the changes in performance with the TER range. The experimental result is present in Table 6, from which we note that the performance on QUAK-M (6.58M) shows an overall improvement compared with those on QUAK-M (100K), and mainly improves between 0.7–1.0. The highest increase for QUAK-M (1.58M) is also seen between 0.7–1.0. As shown in Figure 1, QUAK-M is mainly distributed in the high TER range. Although both QUAK-M (1.58M) and QUAK-M (6.58M) show performance gains over 100K, QUAK-M (6.58M) reports that the performance improvement is not significant at the relatively low TER. We analyze that this in turn, leads to performance degradation of the integrated score compared with 1.58M. This is supported by the fact that even in the TER range of 0.1–0.2, the performance fluctuation of QUAK-M (6.58M) is remarkably lower than that of QUAK-M (1.58M).

In QUAK-P, the amount of data is the lowest at 0.0–0.2 and 0.8–1.0. Therefore, when adding data, the performance variation also shows a lower increase compared with other scores in the range of 0.0–0.1 and 0.8–1.0. From the above results, we conclude that the amount of data can be a contributing factor for performance improvement.

6 Conclusion

We expose three drawbacks in terms of manual QE data construction: human labor and time cost, resulting in limited amount of data and limited language pairs. Taking this into account, we present QUAK, a synthetic Korean-English QE dataset for word-level QE. We automatically generated three sub-QUAKs with three strategies and quantitatively analyzed the trained QE models using them. First, QUAK-P is generated based on parallel corpus and induced the best performance among three sub-QUAKs. Along with QUAK-P, an increase in the data size of QUAK-M had a positive effect on performance gain. However, in further expansion using Wikipedia, the improvement in the low TER range was poor, so the overall performance fell. The QE model trained with QUAK-H did not show a steady performance gain.

This dataset was built in a fully automated manner, eliminating human intervention while increasing reusability and scalability. The QUAK dataset generation process is language-agnostic if there is an MT model and corresponding corpus (monolingual or parallel).

7 Acknowledgements

This work was supported by Institute of Information communications Technology Planning Evaluation(IITP) grant funded by the Korea government(MSIT) (No. 2020-0-00368, A Neural-Symbolic Model for Knowledge Acquisition and Inference Techniques), and supported by the MSIT, Korea, under the Information Technology Research Center(ITRC) support program(IITP-2022-2018-0-01405) supervised by the IITP, and supported by Basic Science Research Program through the National Research Foundation of Korea(NRF) funded by the Ministry of Education(NRF-2021R1A6A1A03045425).

References

Fernando Alva-Manchego, Abiola Obamuyide, Amit Gajbhiye, Frédéric Blain, Marina Fomicheva, and Lucia Specia. 2021. deepquest-py: Large and distilled models for quality estimation. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing: System Demonstrations*, pages 382–389.

John Blatz, Erin Fitzgerald, George Foster, Simona Gandrabur, Cyril Goutte, Alex Kulesza, Alberto Sanchis, and Nicola Ueffing. 2004. Confidence estimation for machine translation. In *Coling 2004: Proceedings of the 20th international conference on computational linguistics*, pages 315–321.

Tom B Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, et al. 2020. Language models are few-shot learners. *arXiv preprint arXiv:2005.14165*.

Davide Chicco and Giuseppe Jurman. 2020. The advantages of the matthews correlation coefficient (mcc) over f1 score and accuracy in binary classification evaluation. *BMC genomics*, 21(1):1–13.

Alexis Conneau, Kartikay Khandelwal, Naman Goyal, Vishrav Chaudhary, Guillaume Wenzek, Francisco Guzmán, Edouard Grave, Myle Ott, Luke Zettlemoyer, and Veselin Stoyanov. 2019. Unsupervised cross-lingual representation learning at scale. *arXiv preprint arXiv:1911.02116*.

Shuoyang Ding, Marcin Junczys-Dowmunt, Matt Post, Christian Federmann, and Philipp Koehn. 2021a. The jhu-microsoft submission for wmt21 quality estimation shared task.

Shuoyang Ding, Marcin Junczys-Dowmunt, Matt Post, and Philipp Koehn. 2021b. Levenshtein training for word-level quality estimation. *arXiv preprint arXiv:2109.05611*.

Chris Dyer, Victor Chahuneau, and Noah A Smith. 2013. A simple, fast, and effective reparameterization of ibm model 2. In *Proceedings of the 2013 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 644–648.

Sugyeong Eo, Chanjun Park, Hyeonseok Moon, Jaehyung Seo, and Heuiseok Lim. 2021a. Dealing with the paradox of quality estimation. In *Proceedings of the 4th Workshop on Technologies for MT of Low Resource Languages (LoResMT2021)*, pages 1–10, Virtual. Association for Machine Translation in the Americas.

Sugyeong Eo, Chanjun Park, Jaehyung Seo, Hyeonseok Moon, and Heuiseok Lim. 2021b. Study on zero-shot based quality estimation. *Journal of the Korea Convergence Society*, 12(11):35–43.

Thierry Etchegoyhen, Eva Martínez Garcia, and Andoni Azpeitia. 2018. Supervised and unsupervised minimalist quality estimators: Vicomtech's participation in the wmt 2018 quality estimation task. In *Proceedings of the Third Conference on Machine Translation: Shared Task Papers*, pages 782–787.

Marina Fomicheva, Lucia Specia, and Francisco Guzmán. 2020a. Multi-hypothesis machine translation evaluation. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 1218–1232.

Marina Fomicheva, Shuo Sun, Erick Fonseca, Chrysoula Zerva, Frédéric Blain, Vishrav Chaudhary, Francisco Guzmán, Nina Lopatina, Lucia Specia, and André FT Martins. 2020b. Mlqe-pe: A multilingual quality estimation and post-editing dataset. *arXiv preprint arXiv:2010.04480*.

Marina Fomicheva, Shuo Sun, Lisa Yankovskaya, Frédéric Blain, Francisco Guzmán, Mark Fishel, Nikolaos Aletras, Vishrav Chaudhary, and Lucia Specia. 2020c. Unsupervised quality estimation for neural machine translation. *Transactions of the Association for Computational Linguistics*, 8:539–555.

Atsushi Fujita and Eiichiro Sumita. 2017. Japanese to english/chinese/korean datasets for translation quality estimation and automatic post-editing. In *Proceedings of the 4th Workshop on Asian Translation (WAT2017)*, pages 79–88.

Amit Gajbhiye, Marina Fomicheva, Fernando Alva-Manchego, Frédéric Blain, Abiola Obamuyide, Nikolaos Aletras, and Lucia Specia. 2021. Knowledge distillation for quality estimation. *arXiv preprint arXiv:2107.00411*.

Hyun Kim, Jong-Hyeok Lee, and Seung-Hoon Na. 2017. Predictor-estimator using multilevel task learning with stack propagation for neural quality estimation. In *Proceedings of the Second Conference on Machine Translation*, pages 562–568.

Philipp Koehn, Hieu Hoang, Alexandra Birch, Chris Callison-Burch, Marcello Federico, Nicola Bertoldi, Brooke Cowan, Wade Shen, Christine Moran, Richard Zens, et al. 2007. Moses: Open source toolkit for statistical machine translation. In *Proceedings of the 45th annual meeting of the association for computational linguistics companion volume proceedings of the demo and poster sessions*, pages 177–180.

Taku Kudo and John Richardson. 2018. Sentencepiece: A simple and language independent subword tokenizer and detokenizer for neural text processing. *arXiv preprint arXiv:1808.06226*.

Guillaume Lample and Alexis Conneau. 2019. Cross-lingual language model pretraining. *arXiv preprint arXiv:1901.07291*.

Dongjun Lee. 2020. Two-phase cross-lingual language model fine-tuning for machine translation quality estimation. In *Proceedings of the Fifth Conference on Machine Translation*, pages 1024–1028.

Yinhan Liu, Jiatao Gu, Naman Goyal, Xian Li, Sergey Edunov, Marjan Ghazvininejad, Mike Lewis, and Luke Zettlemoyer. 2020. Multilingual denoising pre-training for neural machine translation. *Transactions of the Association for Computational Linguistics*, 8:726–742.

Matteo Negri, Marco Turchi, Rajen Chatterjee, and Nicola Bertoldi. 2018. Escape: a large-scale synthetic corpus for automatic post-editing. *arXiv preprint arXiv:1803.07274*.

Myle Ott, Sergey Edunov, Alexei Baevski, Angela Fan, Sam Gross, Nathan Ng, David Grangier, and Michael Auli. 2019. fairseq: A fast, extensible toolkit for sequence modeling. *arXiv preprint arXiv:1904.01038*.

Tharindu Ranasinghe, Constantin Orasan, and Ruslan Mitkov. 2021. An exploratory analysis of multilingual word-level quality estimation with cross-lingual transformers. *arXiv preprint arXiv:2106.00143*.

Matthew Snover, Bonnie Dorr, Richard Schwartz, Linnea Micciulla, and John Makhoul. 2006. A study of translation edit rate with targeted human annotation. In *Proceedings of association for machine translation in the Americas*, volume 200. Citeseer.

Lucia Specia, Frédéric Blain, Marina Fomicheva, Chrysoula Zerva, Zhenhao Li, Vishrav Chaudhary, and André Martins. 2021. Findings of the wmt 2021 shared task on quality estimation. Association for Computational Linguistics.

Lucia Specia, Nicola Cancedda, and Marc Dymetman. 2010. A dataset for assessing machine translation evaluation metrics. In *LREC*.

Lucia Specia, Kashif Shah, Jose G.C. de Souza, and Trevor Cohn. 2013. QuEst - a translation quality estimation framework. In *Proceedings of the 51st Annual Meeting of the Association for Computational Linguistics: System Demonstrations*, pages 79–84, Sofia, Bulgaria. Association for Computational Linguistics.

Lucia Specia, Marco Turchi, Nicola Cancedda, Marc Dymetman, and Nello Cristianini. 2009. Estimating the sentence-level quality of machine translation systems. In *13th Conference of the European Association for Machine Translation*, pages 28–37.

Shuo Sun, Marina Fomicheva, Frédéric Blain, Vishrav Chaudhary, Ahmed El-Kishky, Adithya Renduchintala, Francisco Guzmán, and Lucia Specia. 2020. An exploratory study on multilingual quality estimation. Association for Computational Linguistics.

Yi-Lin Tuan, Ahmed El-Kishky, Adithya Renduchintala, Vishrav Chaudhary, Francisco Guzmán, and Lucia Specia. 2021. Quality estimation without human-labeled data. *arXiv preprint arXiv:2102.04020*.

Minghan Wang, Hao Yang, Hengchao Shang, Daimeng Wei, Jiaxin Guo, Lizhi Lei, Ying Qin, Shimin Tao, Shiliang Sun, Yimeng Chen, et al. 2020. Hw-tsc's participation at wmt 2020 quality estimation shared task. In *Proceedings of the Fifth Conference on Machine Translation*, pages 1056–1061.

Thomas Wolf, Lysandre Debut, Victor Sanh, Julien Chaumond, Clement Delangue, Anthony Moi, Pierric Cistac, Tim Rault, Rémi Louf, Morgan Funtowicz, et al. 2019. Huggingface's transformers: State-of-the-art natural language processing. *arXiv preprint arXiv:1910.03771*.