KIT-19: A Comprehensive Korean Instruction Toolkit on 19 Tasks for Fine-Tuning Korean Large Language Models

Dongjun Jang, Sungjoo Byun, Hyemi Jo, Hyopil Shin

Department of Linguistics, Seoul National University

3-327, 1, Gwanak-ro, Gwanak-gu, Seoul, Republic of Korea

{qwer4107, byunsj, huimei6361, hpshin}@snu.ac.kr

Abstract

Instruction Tuning on Large Language Models is an essential process for model to function well and achieve high performance in specific tasks. Accordingly, in mainstream languages such as English, instruction-based datasets are being constructed and made publicly available. In the case of Korean, publicly available models and datasets all rely on using the output of ChatGPT or translating datasets built in English. In this paper, We introduce *KIT-19* as an instruction dataset for the development of LLM in Korean. *KIT-19* is a dataset created in an instruction format, comprising 19 existing open-source datasets for Korean NLP tasks. In this paper, we train a Korean Pretrained LLM using *KIT-19* to demonstrate its effectiveness. The experimental results show that the model trained on *KIT-19* significantly outperforms existing Korean LLMs. Based on the its quality and empirical results, this paper proposes that *KIT-19* has the potential to make a substantial contribution to the future improvement of Korean LLMs' performance.

Keywords: Large Language Model, Korean Instruction Dataset, Korean LLM Toolkit, Instruction Tuning

1. Introduction

Pretrained LLM (Large Language Models, Shanahan (2022); Brown et al. (2020); Taylor et al. (2022); Chowdhery et al. (2022); Touvron et al. (2023) are Transformer-based language models (Vaswani et al., 2017) with an extensive parameter count, typically in the hundreds of billions or beyond, and they undergo training on vast corpora. These models are typically employed for their intended purposes following instruction tuning methodology, a technique aimed at optimizing models to effectively adhere to user-provided instructions.

Since the release of InstructGPT (Ouyang et al., 2022), instruction datasets in English have been widely developed and made publicly available across various domains such as Ouyang et al. (2022); Taori et al. (2023a). However, for Korean, which falls into the category of a relatively low-resource language, there is a notable absence of datasets created in the native language. Notably, the Korean instruction datasets currently available are either created by translating existing English datasets using DeepL[1] or by relying on outputs from large language models like the ChatGPT API[2], without adequately capturing the cultural nuances of the Korean language (Table 1). Hence, to enhance and advance the performance of Korean Large Language Models, it is imperative to have datasets constructed specifically in the Korean language.

We introduce *KIT-19* (A Comprehensive Korean Instruction Toolkit on 19 Tasks), a universal dataset for Korean Instruction Tuning. *KIT-19* is a dataset constructed for Instruction Tuning, derived from 19 different NLP datasets in the Korean language, each consisting of 5,000 examples. It follows the methodology of Longpre et al. (2023); Bach et al. (2022) which integrates existing NLP datasets into an Instruction Dataset, without relying on machine-translated outputs from other languages or utilizing LLM output as a training dataset. *KIT-19*, as compared to the datasets currently employed for Korean LLM modeling, effectively captures the cultural features of the Korean language and, due to the comprehensive nature of its 19 distinct datasets, contributes to the model's generalizability.

In this paper, we transparently disclose the construction process of *KIT-19* and provide detailed information about the source of the 19 datasets, respectively. To assess the quality of the *KIT-19* dataset, we conduct Full Fine-tuning of our dataset on Polyglot-Ko-5.8b and Polyglot-Ko-1.3b Ko et al. (2023), which are the Korean representative Pretrained LLMs.

Finally, we evaluate Korean Large Language Models publicly available on a total of 6 benchmark datasets for assessment. The results show that the performance of the Polyglot-Ko-5.8b model trained with *KIT-19* outperforms others, and we could also observe that the Polyglot-Ko-1.3b model exhibits higher performance compared to other Korean LLMs.

The contributions of our study are as follows:

- We construct and release 100K Korean instruction datasets, addressing the data scarcity problem and reducing the reliance on translated or GPT-generated instructions for Korean

---

[1] https://www.deepl.com/translator
[2] https://openai.com/blog/chatgpt

LLMs.

- We demonstrate the efficacy of our *KIT-19* by evaluating LLMs using various benchmarks, comparing those trained with *KIT-19* and those without.

2. Related Work

Instruction tuning has emerged as a potent method when pre-trained language models are further trained to comply with specific user-defined instructions. Wei et al. (2022) presented a mixture of over 60 NLP datasets expressed via natural language instructions. Taori et al. (2023b) released 52K instruction data used to fine-tune the Alpaca model. Additionally, Mishra et al. (2022) encapsulated instructions constructed by researchers to crowdsource pre-existing NLP datasets, encompassing 61 unique NLP tasks and a total of 193k instances. The dataset presented by Sanh et al. (2021) is the outcome of integrating 170 English NLP datasets and 2,052 English prompts for the purpose of instruction fine-tuning. Other noteworthy open-source instruction datasets in English include UnifiedQA (Khashabi et al., 2020), LIMA (Zhou et al., 2023), and those generated using Chat-GPT4 (Peng et al., 2023).

While multilingual instructions, exemplified by Wang et al. (2022), are in existence, open-source instruction datasets for languages other than English remain limited in number. Focusing on Korean, the Koalpaca dataset[25] adopts translations from DeepL to render Alpaca instructions in the Korean language. The instruction set used for KULLM training Lab and research (2023) draws its foundation from English datasets, specifically Peng et al. (2023), Zheng et al. (2023), and Conover et al. (2023), translating them into Korean. Hahn (2023) follows a similar path, presenting a Korean translation of the LIMA dataset Zhou et al. (2023). Datasets like Counsel GPT[26] and Human Rights Corpus (HRC) (Song et al., 2023) are crafted using Chat-GPT[27] and are original Korean instructions, not relying on translations. However, these are specialized datasets and include fewer than 1.5K instances each.

3. KIT-19 (A Comprehensive Korean Instruction Toolkit on 19 Tasks)

Most Korean Instruction datasets rely on translating from English datasets or using the output of high-performing models like ChatGPT. Table 1 lists the currently available open-source Korean instructions and provides their construction methods[28]. All the Korean instruction datasets mentioned above, except for the namuwiki_alpaca_dataset, rely on machine translation or the ChatGPT generation. Furthermore, each dataset lacks a comprehensive set of tasks, which means that there's a potential for the model to become biased toward specific tasks during the Instruction Tuning process. While it may be acceptable for a model designed solely for one task, for creating models capable of performing a wide range of comprehensive tasks, a comprehensive instruction dataset is required.

Accordingly, in our research, we introduce *KIT-19*[29] (A Comprehensive Korean Instruction Toolkit on 19 Tasks), which integrates the 19 existing open-source NLP datasets as the instruction dataset in Korean. We collected 5000 instances for each of the 19 tasks, summing up to a total of $100K$ instances. Notably, one of these 19 datasets, kowiki_text is seperated in two distinct categories, which is why the overall instance count is $100K$. Following the approach proposed by Wei et al. (2022) and Raffel et al. (2023), we collected each dataset within *KIT-19* with a ratio of 5,000 instances per task to ensure that the dataset does not exhibit bias toward specific tasks. Thus, we restrict the number of instances from each dataset to 5,000. Figure 1 displays the source NLP datasets and the categories within our extensive instruction dataset.

The format of *KIT-19* follows that of Taori et al. (2023b), including *Instruction, Input,* and *Output*. Instruction describes the task the model should perform. Input is an optional context or input for the task. For example, when the instruction is 'Tell me the definition of the term', the terminology should be the input. Output is an required response that we expect the model to generate. Figure 2 shows the construction procedure of *KIT-19*.

We categorize the 19 tasks comprising *KIT-19* into 10 top-level tasks and introduce the methodology for transforming the objectives of each dataset into an Instruction Dataset (three primary components: Instruction, Input, and Output). In this section, we hope that our explanations would be beneficial for future works in constructing Instruction datasets.

3.1. Hate Speech Detection

Hate Speech Detection is the task of determining whether an input sentence contains hate speech or derogatory expressions. Relying solely on trans-

---

Dataset	Construction Method
KoAlpaca v1.0[3]	After **translating** the Alpaca instruction, the output is generated by **ChatGPT**.
KoAlpaca v1.1[4]	After collecting questions from JiSikIn[5] (Korean online knowledge-sharing platform), answers were generated using **ChatGPT**.
sharegpt_deepl_ko[6]	**Translated** the ShareGPT data using DeepL.
ShareGPT-74k-ko[7]	**Translated** the cleaned version of ShareGPT with 90k using Google Translate.
KoChatGPT[8]	After collecting questions from the Korean question dataset, answers were generated using **ChatGPT**.
OIG-small-chip2-ko[9]	**Translated** the OIG-smallchip-2[10] English data from LAION AI using **Google Translate**.
Korquad-Chat[11]	Given the context (paragraphs from news and Wikipedia) of the KorQuAD v1 data, relevant conversations were generated using **ChatGPT**.
AIRC-KETI/kowow[12]	The **translated** data of WoW (Wizard of Wikipedia) - a knowledge-based conversation dataset.
CounselGPT[13]	Counseling data generated by **ChatGPT API**.
Evolve-instruct[14]	Instructions were augmented using evol-instruct from WizardLM, and then answers were generated by **ChatGPT**.
KULLM v2[15]	**Translated** the GPT4ALL, Dolly, and Vicuna (ShareGPT) data using DeepL.
namuwiki_alpaca_dataset[16]	A dataset modified to fit Stanford Alpaca training from the NamuWiki (Korean wiki platform) dump file.
ko-lima-vicuna[17]	A dataset regenerated in Korean using the **ChatGPT API**, based on the lima_vicuna_format[18] data.
ko-lima[19]	A dataset **translated** into Korean from the training data of Zhou et al. (2023)
Ko-StrategyQA[20]	This dataset is the Korean version of StrategyQA[21]. All questions and paragraphs from the original dataset were **translated** using DeepL.
KOpen-platypus[22]	**Translated** Lee et al. (2023) with DeepL.
EverythingLM-data-V2-Ko[23]	**Translated** EverythingLM V2[24] in DeepL.
human-rights-corpus/HRC/	Using the precedents and consultation cases from the National Human Rights Commission of South Korea as a reference, **GPT-3.5-turbo** for one-shot learning was employed to generate question-answer pairs.

Table 1: Existing Korean Instruction Datasets and Construction Methods: It is problematic that most of the existing instruction datasets rely on translation and ChatGPT.

lated versions of this task is ineffective, as hate speech is deeply rooted in a language's unique cultural, political, and linguistic nuances. It is also noteworthy that closed generative models such as ChatGPT and CLOVA X [30] do not process or produce unethical content, thereby hindering the generation of instruction datasets for this task. We generate hate speech detection instruction datasets from three existing NLP datasets: APEACH (Yang et al., 2022), UnSmile (Kim, 2022), and hatescore (Kang et al., 2022).

- **Instruction**: The instructions primarily focus

on questions pertaining to hate speech detection. However, the design of these instructions varies depending on the specific source dataset. Depending on the unique attributes of the original data (for example, whether the dataset categorizes content simply as containing hate speech or not, or if it annotates the target of the hate speech), the instructions might be framed as 'Does this {Input} contain offensive content?' or 'Who or what is the target of this hate speech?'.

- **Input**: The input for this task corresponds to the online comments provided by APEACH, UnSmile, and hatescore datasets.

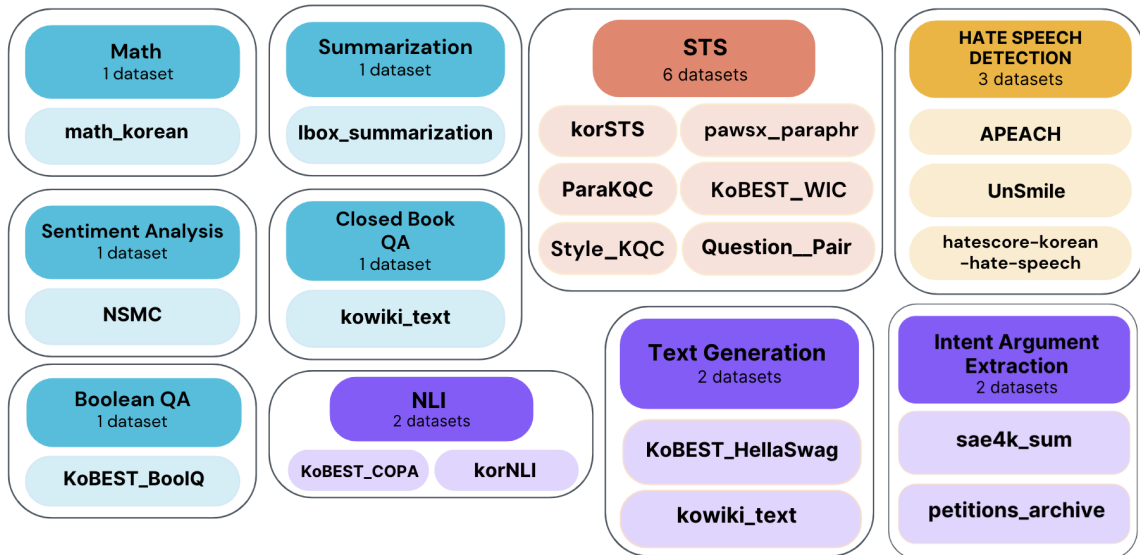[30] https://clova-x.naver.com/welcome

Figure 1: A glance at the KIT for Korean LLM: We create instruction datasets by drawing from 19 Korean NLP datasets across 10 different categories. We utilize 'kowiki_text' as a source dataset for both Closed Book QA and Next Sentence Prediction tasks.
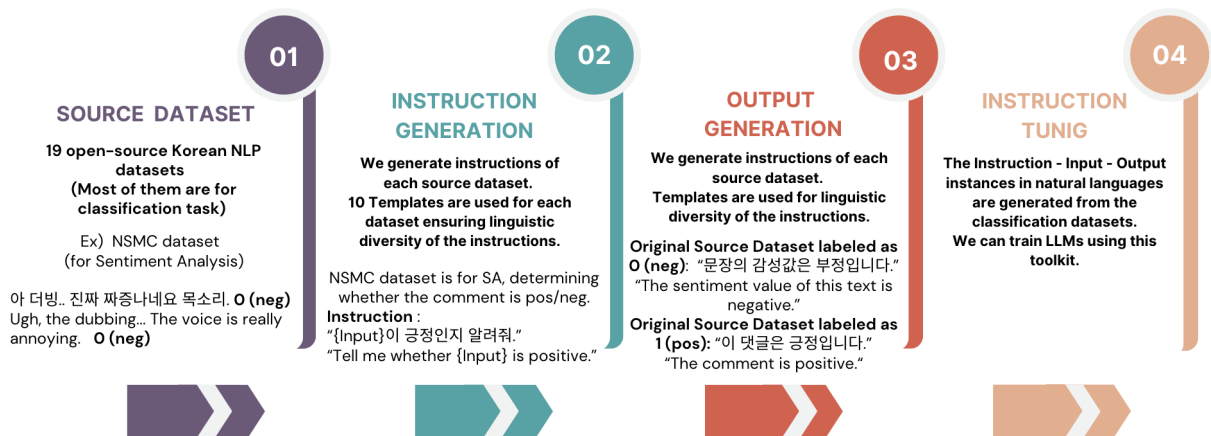

Figure 2: Overview of the data construction procedure of *KIT-19*

- **Output**: The output is tailored based on the characteristics of each source dataset. For instance, if the original dataset categorizes content in a binary manner, determining whether it contains hate speech, the output is phrased as 'The comment is hate speech.'. On the other hand, if the source dataset provides annotations on the target of the hate speech, the output is framed as 'The text contains derogatory language aimed at LGBT.'.

3.2. Boolean Question Answering (QA)

Boolean QA requires the model to respond with either a 'Yes' or 'No' based on the provided context. We use the KoBEST_BoolQ (Kim et al., 2022) to generate instructions, input, and output.

- **Instruction**: The instruction is typically

phrased as a question, such as 'Is the question or the statement in the Input True or False?'

- **Input**: For this task, we use paragraphs and corresponding questions from the KoBEST_BoolQ dataset. Example is as follows:

Paragraph: The term 'French Revolution' mainly refers to the 1789 revolution, although it technically encompasses both the July Revolution of 1830 and the February Revolution of 1848. The 1789 event often stands out and is termed the 'Great French Revolution.'

Question: Isn't the July Revolution part of the Great French Revolution?

- **Output**: The output provides a clear True or False response based on the context in the

paragraph. For instance, the response might be, 'Based on the paragraph, the answer to your question is False.'

3.3. Natural Language Inference (NLI)

In the NLI task, it is essential to discern the semantic relationship between a given 'hypothesis' and 'premise' to determine if they entail, contradict, or are neutral with respect to each other. We derive our NLI instructions from the KoBEST_COPA (Kim et al., 2022) and korNLI (Ham et al., 2020) datasets. Outputs based on KoBEST_COPA categorize the relationship as cause and effect between two sentences. In contrast, those based on korNLI indicate if the sentences are in entailment, contradiction, or neutral relation. The guiding question for this task is 'What is the relationship between the two provided sentences?'.

- **Instruction**: The questions mainly address the semantic relationship between two input sentences. For instance, 'How are two sentences, {Input}, semantically related?'.

- **Input**: The paired sentences from the KoBEST_COPA and korNLI datasets are the input for this task.

- **Output**: Outputs are designed according to the specifics of each source dataset. For example, using the KoBEST_COPA dataset, the output might state 'Sent 1 is the cause and Sent 2 is the effect.', whereas with the korNLI dataset, it might say 'These sentences contradict each other.'

3.4. Text Generation

Text Generation involves creating a sentence or a paragraph that logically follows an input text. We transform two existing datasets into instruction sets for this Text Generation task: KoBEST_HellaSwag (Kim et al., 2022) and kowiki_text [31]. For KoBEST_HellaSwag, the output is a sentence that naturally succeeds the input text. In contrast, for kowiki_text, the output is a full paragraph. The kowiki_text dataset is primarily based on the Korean Wikipedia. For data generation, we split the text in half, using the first part as input and the latter half as output.

- **Instruction**: The primary prompt for this task is 'Produce text that naturally continues from the presented text.'

- **Input**: For KoBEST_HellaSwag, the preceding sentence from the original dataset serves as the input. Meanwhile, for kowiki_text, we use the first section of the divided Wikipedia text.

- **Output**: The output is the succeeding sentence or paragraph drawn from the source datasets.

3.5. Semantic Textual Similarity (STS)

Semantic Textual Similarity (STS) is a task that measures the semantic similarity between two sentences or texts. For this task, we source from 6 datasets: korSTS (Ham et al., 2020), pawsx_paraphr (Yang et al., 2019), ParaKQC (Cho et al., 2020), KoBEST_WIC (Kim et al., 2022), Style_KQC (Cho et al., 2022), and Question_Pair [32].

- **Instruction**: A typical instruction for the STS task might ask, 'Do these sentences convey the same meaning?'

- **Input**: We utilize pairs of sentences or texts from the previously mentioned six datasets as our input.

- **Output**: The output evaluates how similar in meaning the input sentences are. A possible response could be, 'Yes, the sentences are semantically alike.' However, for the korSTS dataset, which provides a quantified similarity score ranging from 0 to 5, an output might be phrased as, 'The similarity between the two provided texts scores 4.2 out of 5.'

3.6. Sentiment Analysis (SA)

Sentiment Analysis (SA) determines if a text has a positive or negative tone. Generating instruction datasets for this task is challenging if machine translation or ChatGPT are used since it is difficult to process euphemisms, typos, or profanities commonly found in online comments. The Naver Sentiment Movie Corpus, namely NSMC dataset (Park, 2016) consists of movie comments[33], which are labeled either 0 for negative or 1 for positive sentiments. Rather than translating English SA instruction datasets or generating instructions with ChatGPT, we use NSMC datasets to generate three components as follows.:

- **Instruction**: This is a textual string, asking to determine the sentiment value of the input sentence, such as 'Is the provided text positive or negative?'

- **Input**: Our input consists of movie reviews sourced from the NSMC dataset.

---

[31] https://github.com/lovit/kowikitext

[32] https://github.com/songys/Question_pair

[33] http://movie.naver.com/movie/point/af/list.nhn

Input Sentences

우리 인터넷 통신사 좀 알려줘.
Tell me our internet service provider.

우리 인터넷 가입통신사 확인 좀 해줘라
Please check our internet subscription service provider for me.

Output

두 문장은 같은 뜻입니다.
The two sentences convey the same meaning.

Template

Template #1
###질문: "**{Input}** 이 서로 유사한가요?",
"**{Output}**"
###Question: " Are **{Input}** semantically similar.", "**{Output}**"

...

Template #10
질문: "**{Input}**이 사실상 같은 말을 하고 있는지 알려주세요."\n\n, "**{output}**
###Question: "Please tell me if **{Input}** are essentially saying the same thing."\n\n
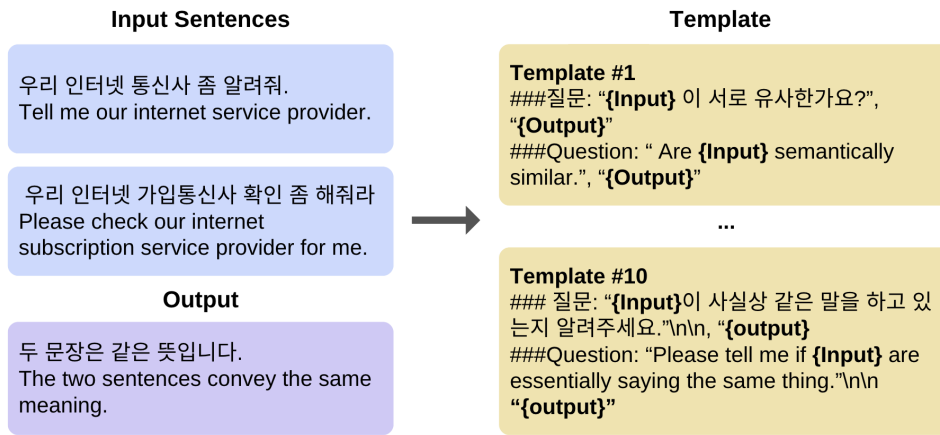**"{output}"**

Figure 3: Instruction Template used to construct *KIT-19*. Above is the example of one of the datasets used for STS task. We employ 10 unique templates for each dataset, resulting in a total of 200 templates.

- **Output**: Instead of binary responses like 0 or 1, we phrase the outputs in natural language, such as 'The provided text has a negative tone.'

3.7. Intent Argument Extraction

Intent Argument Extraction is aimed at discerning the primary purpose or intent embedded within a text. We leverage two NLP datasets for guiding this category: sae4k_sum Cho et al. (2018) and petitions_archive [34]. sae4k_sum comprises *question–command–intent* pairs and the labels for classification of the utterance type. petitions_archive provides data sourced from public petitions presented to the Blue House (Cheong Wa Dae). This dataset offers both the content of these petitions and their respective titles. We interpret the title as encapsulating the core intent of the petition. As such, when crafting outputs, we utilize the title segment.

- **Instruction**: For this particular task, the guiding question is: 'What is the author's intent behind the text?'

- **Input**: We draw from the question/command portion of the sae4k_sum dataset and the main text from the petitions_archive dataset for our inputs.

- **Output**: We use the intent segment from the sae4k_sum dataset as the target output. For the petitions_archive dataset, its title serves as the output to indicate intent.

3.8. Math

This task requires the model to provide accurate answers to mathematical problems. We create

mathematical instructions using the math_korean dataset[35].

The original dataset is in JSON format and includes key information such as class (the type of mathematical question, e.g., Size comparison or Arithmetic operations), question (the math problem itself), answer, and an equation. We transform this JSON dataset into three components: instruction, input, and output.

- **Instruction**: This is a textual string, for example, 'What is the correct answer to this math problem?'

- **Input**: This corresponds to the question from the original dataset. For instance, 'Students in Shinhye's class surveyed their favorite sports among the students. Among those surveyed, the ratio of students who like swimming and soccer is (2/5) of those who like swimming, and the students who like soccer are twice as many as those who like swimming. If there are 10 students who like swimming, how many students like soccer?'

- **Output**: The output is structured as follows: 'The answer is 8, as calculated by 10 * (2/5) * 2.' Both the answer and the equation from the original JSON dataset are utilized in generating the output.

3.9. Closed Book QA

Closed Book QA involves a model answering open-domain questions without relying on external knowledge. In this task, we create instructions using the kowiki_text dataset[36]. While this dataset has previously been used in Text Generation tasks, we

---

[34]https://github.com/lovit/petitions_archive

[35]https://github.com/jkc-ai/mwp-korean-data-2021

[36]https://github.com/lovit/kowikitext

	Metric	Polyglot-ko-1.3b	Polyglot-ko-5.8b	KoAlpaca-5.8B	kullm-polyglot-5.8b-v2	KIT-5.8b	KIT-1.3b
KoBEST_COPA	ACC	72.00%	77.60%	69.80%	76.60%	**91.60%**	83.80%
	F1 (macro)	71.96%	77.55%	69.77%	76.53%	**91.59%**	83.78%
KoBEST_BoolQ	ACC	49.86%	53.63%	56.34%	50.28%	**66.24%**	50.71%
	F1 (macro)	35.52%	43.56%	50.64%	33.71%	**66.14%**	34.78%
KoBEST_HellaSwag	ACC	40.60%	48.80%	38.20%	44.40%	**97.60%**	81.60%
	ACC_Norm	53.00%	59.80%	46.20%	55.20%	**98.20%**	89.80%
	F1 (macro)	40.13%	48.53%	38.15%	44.25%	**97.61%**	81.49%
KLUE_STS	ACC	42.39%	45.28%	51.83%	42.39%	**65.51%**	42.20%
	F1	59.54%	60.34%	33.86%	59.54%	**69.71%**	56.52%
KoBEST_SentiNeg	ACC	69.27%	50.38%	38.79%	50.38%	71.54%	**80.86%**
	F1	68.19%	33.95%	38.48%	33.50%	68.98%	**80.86%**
KLUE_YNAT	F1	33.24%	33.62%	20.91%	32.20%	28.15%	**38.34%**

Table 2: Experiment Results: We evaluate 7 models on 6 different benchmark datasets. KIT-5.8b and KIT-1.3b, which use our own templates as shown in Figure 3, out-perform the others. Original templates were used for the other models. Detailed experimental setups and additional results can be found in Appendix A

approach it differently in this task. Instead of splitting the Wikipedia text in half, we use the definitions or descriptions of terminology. The dataset is transformed into three components as follows.

- **Instruction**: This consists of a textual string, such as 'Provide information about {Input}.'

- **Input**: The input for this task corresponds to the title of the wiki text provided by kowiki_text. In other words, it represents the main topic or subject being questioned in the instruction. For instance, 'Ivan Aivazovsky's life' could be an example of input data.

- **Output**: The output is structured based on the content found in kowiki_text. For example, 'Ivan Aivazovsky was born in a small village in Feodosiya, on the Crimean Peninsula of the Russian Empire, to a poor Armenian family. His parents' last name was Aivazian...'

3.10. Summarization

Summarization task requires a model to summarize the key content of the input text. We use lbox_summarization dataset (Hwang et al., 2022) to generate instructions. lbox_summarization dataset consists of 20K *precendent – summary* pairs. We use 5K pairs among them to convert the dataset into instructions, input, and output.

- **Instruction**: This consists of a textual string, such as 'Briefly summarize {Input}.'

- **Input**: The input for this task corresponds to the main precedent of the original lbox_summarization dataset.

- **Output**: The output is structured based on the summary part of the lbox_summarization dataset.

4. 200 Instruction Templates for the Utilization of *KIT-19* During the Training Process

KIT-19 is a Comprehensive Dataset consisting of 10 categories. To fully capture the characteristics of each task, we introduced the process of instructifying each task in Section 3. In Section 3, we have organized 10 templates for each task, enabling the utilization of the constructed datasets for their respective objectives. These templates differ in aspects like sentence structure, the use of honorifics, and phrasing. We craft templates based on the specific features of each source dataset to maintain its distinct features. Furthermore, we diversify the output of each task to ensure that the model can provide a variety of responses. We generate outputs considering the characteristics of the source datasets.

5. Experimental Setup

We believe that *KIT-19* is a well human-crafted dataset as the Korean instruction dataset. When evaluating the quality of the constructed dataset, both the dataset construction process and distribution are essential, but the performance enhancement when applying the dataset to a model is also a critical factor. In this section, we assess the performance of KIT-5.8b and KIT-1.3b models trained on Polyglot-Ko-5.8b and Polyglot-Ko-1.3b models, respectively, with the utilization of *KIT-19*.

For evaluation, we follow the method of Gao et al. (2021). We used the polyglot branch of lm-evaluation-harness to utilize Korean benchmark datasets implemented for evaluation. Models are evaluated on two types of templates: an original template provided by lm-evaluation-harness and a custom template used in training KIT models as

shown in Figure 3. By varying the format of evaluation templates, we examine the extent to which the evaluation score is affected by the accordance of templates used in training and evaluation. We set $batch\_size$ as 32 to facilitate comparison with other Korean LLMs and $num\_fewshot$ as 0, 1, 2, 3 for the ablation study.

5.1. Models and Benchmarks

As baseline models, we test Polyglot-Ko-1.3b, Polyglot-Ko-5.8b, Koalpaca-5.8b[37], and KULLM (Lab and research, 2023) models which are the mostly employed in the Korean NLP fields.

For benchmarks, we use KoBEST_COPA, BoolQ, HellaSwag, SentiNeg (Kim et al., 2022), KLUE_STS, and KLUE_YNAT (Park et al., 2021) to investigate whether the instruction tuning improved the model's performance both on seen tasks and unseen tasks. Among these, SentiNeg, KLUE_STS, and KLUE_YNAT (KLUE-TC) are unseen datasets[38] that are not included in *KIT-19*. We choose (macro) f1 as our evaluation metric to match the guideline presented in the literature and to facilitate comparison with models including polyglot. Every KOBEST benchmark and KLUE_STS present F1 as their proper evaluation metric except KLUE_YNAT which chooses macro F1. We select macro F1 for KOBEST and F1 for KLUE among the evaluation metrics available in the evaluation framework to follow the literature as much as possible.

5.2. Hyper-Parameters

We train the models for 1 epoch, using a batch size of 8 for the KIT-1.3b model and 4 for the KIT-5.8b model, employing 1024 gradient accumulation steps. We set the learning rate at 3e-5. We use four pieces of H-100 GPUs for training process.

6. Results

For the evaluation of *KIT-19*, we select Korean LLMs based on the Polyglot-Ko models. Polyglot-Ko represents pretrained LLMs without undergoing Instruction Tuning. The Kullm-polyglot-5.8-v2 model and KoAlpaca-5.8B model are the instruction tuned models that have undergone Instruction Tuning, each utilizing the Polyglot-Ko-5.8b model as a base, as illustrated in Figure 1. In the case of benchmark datasets, to ensure fairness, we include three of the Unseen Benchmark tasks out of

the six tasks, as detailed in Section 5.1. Table 2 displays the result of the evaluation, showing that the models trained with our KIT-19 outperform in all the benchmark, with $num_{fewshot}$ of the models' highest performance varying in each task.

6.1. The Effectiveness of the *KIT-19* Dataset in Terms of Performance

As shown in Table 2, it is evident that the KIT-5.8b[39] and KIT-1.3b[40] models significantly outperform existing Korean LLMs. When compared to the other four models, it is clear that the KIT models exhibit the highest performance across all six benchmark sets used in the experiments. Notably, the performance gap compared to other models averages at 15% higher, underscoring the data quality of *KIT-19* in terms of performance. This also serves as direct evidence highlighting the limitations of Instruction Datasets relying on translation and ChatGPT.

The KIT-5.8b model exhibited the highest performance in four tasks (three Seen datasets and one Unseen dataset), notably excelling in the three Seen datasets of the KoBEST series compared to other models. While this performance improvement could be attributed to the inclusion of the dataset in the training process, our interpretation is that the task-specific ratios within the *KIT-19* dataset played a crucial role in achieving well-rounded performance enhancements without bias toward particular tasks. The model achieved accuracy of over 90% in the COPA and HellaSwag benchmarks, underscoring the quality and significance of the *KIT-19* dataset.

Finally, the KIT-1.3b model, despite having model parameters over four times smaller than the 5.8b model, outperformed both the KoAlpaca-5.8b and Kullm-Polyglot-5.8b-v2 models. In particular, the KIT-1.3b model exhibited the highest performance in the SentiNeg and YNAT benchmarks. Furthermore, it achieved an accuracy of over 80% in the COPA and HellaSwag benchmark datasets, thus validating the efficacy of the *KIT-19* dataset.

6.2. The Necessity for Expanding the Domains of the *KIT-19* Dataset

KIT-19 has convincingly demonstrated its effectiveness by outperforming other models in the model performance evaluation process. Moreover, the KIT models exhibited superior performance in the Unseen Benchmarks compared to other models, suggesting that *KIT-19* indirectly facilitated learning

---

[37]https://github.com/Beomi/KoAlpaca

[38]Note that KLUE_STS differs from the other two unseen tasks in that one of its source corpora, PARAKQC, overlaps with our KIT-19 dataset, which makes KLUE_STS a partially unseen task.

[39]https://huggingface.co/snunlp/KIT-5.8b/

[40]https://huggingface.co/snunlp/KIT-1.3b/

in the Out-Domain during model training. However, to ensure stable performance in tasks beyond benchmarks, it is essential to include a variety of domains. In our future research, we plan to expand the *KIT-19* to overcome challenges related to Out-Domain scenarios.

7. Conclusion

In this paper, we have addressed the need for native language instruction datasets for Korean LLMs. While Pre-trained LLMs could have robustness with further training, especially when fine-tuned with instruction-style datasets, the availability of instruction datasets in Korean has been notably scarce. Accordingly, we introduce *KIT-19*, filling a critical gap in the development of Korean LLMs. This comprehensive dataset is humanly constructed by integrating 19 existing Korean NLP tasks, each comprised of 5,000 examples. Our work contains the constructing procedure of *KIT-19* for future works, explaining the methodology for transforming the objectives of each tasks into an Instruction Dataset. For each task, we provide 10 templates, allowing them to be used adequately for their specific objectives. Additionally, we diversified the output for each task to ensure the model could generate a variety of responses. For validation of our dataset, we fine-tuned *KIT-19* on two Korean Pre-trained LLMs and evaluated their performance on six benchmark datasets. We also compared the results with those of four other different models. As a result, we observed that our models outperformed other Korean models significantly. This not only highlights the limitations of models relying on translated or LLM-generated datasets but also underscores the essential role of *KIT-19*. Our models exhibited superior performance in the Unseen benchmarks, outperforming other models. Our research team interprets this result as a consequence of indirectly learning unseen information during the training process. For future research, we plan to expand *KIT-19* to include more domains, enhancing the generality of the LLMs.

8. Ethical Statement

In our research, we employed existing NLP datasets to produce our instruction datasets. It is imperative to highlight that each of these datasets comes with permissions for modification and distribution. We have reviewed and ensured the adherence to the licensing and copyright stipulations associated with each dataset.

Releasing our dataset to the public is intended to promote progress in Korean language model development. We emphasize the value of open research and our commitment to enhancing the shared intellectual resources of the NLP field.

9. Bibliographical References

Yuvanesh Anand, Zach Nussbaum, Brandon Duderstadt, Benjamin Schmidt, and Andriy Mulyar. 2023. Gpt4all: Training an assistant-style chatbot with large scale data distillation from gpt-3.5-turbo. https://github.com/nomic-ai/gpt4all.

Stephen H Bach, Victor Sanh, Zheng-Xin Yong, Albert Webson, Colin Raffel, Nihal V Nayak, Abheesht Sharma, Taewoon Kim, M Saiful Bari, Thibault Fevry, et al. 2022. Promptsource: An integrated development environment and repository for natural language prompts. *arXiv preprint arXiv:2202.01279*.

Tom Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared D Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, et al. 2020. Language models are few-shot learners. *Advances in neural information processing systems*, 33:1877–1901.

BSI. 1973a. *Natural Fibre Twines*, 3rd edition. British Standards Institution, London. BS 2570.

BSI. 1973b. Natural fibre twines. BS 2570, British Standards Institution, London. 3rd. edn.

A. Castor and L. E. Pollux. 1992. The use of user modelling to guide inference and learning. *Applied Intelligence*, 2(1):37–53.

J.L. Chercheur. 1994. *Case-Based Reasoning*, 2nd edition. Morgan Kaufman Publishers, San Mateo, CA.

Won Ik Cho, Jong In Kim, Young Ki Moon, and Nam Soo Kim. 2020. Discourse component to sentence (dc2s): An efficient human-aided construction of paraphrase and sentence similarity dataset. In *Proceedings of The 12th Language Resources and Evaluation Conference*, pages 6819–6826.

Won Ik Cho, Sangwhan Moon, Jong In Kim, Seok Min Kim, and Nam Soo Kim. 2022. Stylekqc: A style-variant paraphrase corpus for korean questions and commands.

Won Ik Cho, Young Ki Moon, Woo Hyun Kang, and Nam Soo Kim. 2018. Extracting arguments from korean question and command: An annotated corpus for structured paraphrasing. *arXiv preprint arXiv:1810.04631*.

N. Chomsky. 1973. Conditions on transformations. In *A festschrift for Morris Halle*, New York. Holt, Rinehart & Winston.

Aakanksha Chowdhery, Sharan Narang, Jacob Devlin, Maarten Bosma, Gaurav Mishra, Adam Roberts, Paul Barham, Hyung Won Chung, Charles Sutton, Sebastian Gehrmann, et al. 2022. Palm: Scaling language modeling with pathways. *arXiv preprint arXiv:2204.02311*.

Mike Conover, Matt Hayes, Ankit Mathur, Jianwei Xie, Jun Wan, Sam Shah, Ali Ghodsi, Patrick Wendell, Matei Zaharia, and Reynold Xin. 2023. Free dolly: Introducing the world's first truly open instruction-tuned llm.

Umberto Eco. 1990. *The Limits of Interpretation*. Indian University Press.

Leo Gao, Jonathan Tow, Stella Biderman, Sid Black, Anthony DiPofi, Charles Foster, Laurence Golding, Jeffrey Hsu, Kyle McDonell, Niklas Muennighoff, Jason Phang, Laria Reynolds, Eric Tang, Anish Thite, Ben Wang, Kevin Wang, and Andy Zou. 2021. A framework for few-shot language model evaluation.

Taeseung Hahn. 2023. Ko-lima: Korean lima dataset for efficient instruction-tuning.

Jiyeon Ham, Yo Joong Choe, Kyubyong Park, Ilji Choi, and Hyungjoon Soh. 2020. Kornli and korsts: New benchmark datasets for korean natural language understanding. *arXiv preprint arXiv:2004.03289*.

Paul Gerhard Hoel. 1971a. *Elementary Statistics*, 3rd edition. Wiley series in probability and mathematical statistics. Wiley, New York, Chichester. ISBN 0 471 40300.

Paul Gerhard Hoel. 1971b. *Elementary Statistics*, 3rd edition, Wiley series in probability and mathematical statistics, pages 19–33. Wiley, New York, Chichester. ISBN 0 471 40300.

Wonseok Hwang, Dongjun Lee, Kyoungyeon Cho, Hanuhl Lee, and Minjoon Seo. 2022. A multitask benchmark for korean legal language understanding and judgement prediction.

Otto Jespersen. 1922. *Language: Its Nature, Development, and Origin*. Allen and Unwin.

TaeYoung Kang, Eunrang Kwon, Junbum Lee, Youngeun Nam, Junmo Song, and JeongKyu Suh. 2022. Korean online hate speech dataset for multilabel classification: How can social science improve dataset on hate speech?

Daniel Khashabi, Sewon Min, Tushar Khot, Ashish Sabharwal, Oyvind Tafjord, Peter Clark, and Hannaneh Hajishirzi. 2020. Unifiedqa: Crossing format boundaries with a single qa system.

Dohyeong Kim, Myeongjun Jang, Deuk Sin Kwon, and Eric Davis. 2022. Kobest: Korean balanced evaluation of significant tasks.

Seonghyun Kim. 2022. Korean unsmile dataset: Human-annotated multi-label korean hate speech dataset. https://github.com/smilegate-ai/korean_unsmile_dataset.

Hyunwoong Ko, Kichang Yang, Minho Ryu, Taekyoon Choi, Seungmu Yang, jiwung Hyun, and Sungho Park. 2023. A technical report for polyglot-ko: Open-source large-scale korean language models.

NLP & AI Lab and Human-Inspired AI research. 2023. Kullm: Korea university large language model project. https://github.com/nlpai-lab/kullm.

Ariel N. Lee, Cole J. Hunter, and Nataniel Ruiz. 2023. Platypus: Quick, cheap, and powerful refinement of llms.

Shayne Longpre, Le Hou, Tu Vu, Albert Webson, Hyung Won Chung, Yi Tay, Denny Zhou, Quoc V Le, Barret Zoph, Jason Wei, et al. 2023. The flan collection: Designing data and methods for effective instruction tuning. *arXiv preprint arXiv:2301.13688*.

Swaroop Mishra, Daniel Khashabi, Chitta Baral, and Hannaneh Hajishirzi. 2022. Cross-task generalization via natural language crowdsourcing instructions.

Long Ouyang, Jeffrey Wu, Xu Jiang, Diogo Almeida, Carroll Wainwright, Pamela Mishkin, Chong Zhang, Sandhini Agarwal, Katarina Slama, Alex Ray, et al. 2022. Training language models to follow instructions with human feedback. *Advances in Neural Information Processing Systems*, 35:27730–27744.

Lucy Park. 2016. Naver sentiment movie corpus. urlhttps://github.com/e9t/nsmc.

Sungjoon Park, Jihyung Moon, Sungdong Kim, Won Ik Cho, Jiyoon Han, Jangwon Park, Chisung Song, Junseong Kim, Yongsook Song, Taehwan Oh, Joohong Lee, Juhyun Oh, Sungwon Lyu, Younghoon Jeong, Inkwon Lee, Sangwoo Seo, Dongjun Lee, Hyunwoo Kim, Myeonghwa Lee, Seongbo Jang, Seungwon Do, Sunkyoung Kim, Kyungtae Lim, Jongwon Lee, Kyumin Park, Jamin Shin, Seonghyun Kim, Lucy Park, Alice

Oh, Jungwoo Ha, and Kyunghyun Cho. 2021. Klue: Korean language understanding evaluation.

Baolin Peng, Chunyuan Li, Pengcheng He, Michel Galley, and Jianfeng Gao. 2023. Instruction tuning with gpt-4.

Colin Raffel, Noam Shazeer, Adam Roberts, Katherine Lee, Sharan Narang, Michael Matena, Yanqi Zhou, Wei Li, and Peter J. Liu. 2023. Exploring the limits of transfer learning with a unified text-to-text transformer.

Victor Sanh, Albert Webson, Colin Raffel, Stephen H. Bach, Lintang Sutawika, Zaid Alyafeai, Antoine Chaffin, Arnaud Stiegler, Teven Le Scao, Arun Raja, Manan Dey, M Saiful Bari, Canwen Xu, Urmish Thakker, Shanya Sharma Sharma, Eliza Szczechla, Taewoon Kim, Gunjan Chhablani, Nihal Nayak, Debajyoti Datta, Jonathan Chang, Mike Tian-Jian Jiang, Han Wang, Matteo Manica, Sheng Shen, Zheng Xin Yong, Harshit Pandey, Rachel Bawden, Thomas Wang, Trishala Neeraj, Jos Rozen, Abheesht Sharma, Andrea Santilli, Thibault Fevry, Jason Alan Fries, Ryan Teehan, Stella Biderman, Leo Gao, Tali Bers, Thomas Wolf, and Alexander M. Rush. 2021. Multitask prompted training enables zero-shot task generalization.

Murray Shanahan. 2022. Talking about large language models. *arXiv preprint arXiv:2212.03551*.

Charles Joseph Singer, E. J. Holmyard, and A. R. Hall, editors. 1954–58. *A history of technology*. Oxford University Press, London. 5 vol.

Young-Sook Song, Sang-Jin Shim, and Sung-Hyun Kim. 2023. Construction of a human rights corpus for conversational generation models.

Jannik Strötgen and Michael Gertz. 2012. Temporal tagging on different domains: Challenges, strategies, and gold standards. In *Proceedings of the Eight International Conference on Language Resources and Evaluation (LREC'12)*, pages 3746–3753, Istanbul, Turkey. European Language Resource Association (ELRA).

S. Superman, B. Batman, C. Catwoman, and S. Spiderman. 2000. *Superheroes experiences with books*, 20th edition. The Phantom Editors Associates, Gotham City.

Rohan Taori, Ishaan Gulrajani, Tianyi Zhang, Yann Dubois, Xuechen Li, Carlos Guestrin, Percy Liang, and Tatsunori B Hashimoto. 2023a. Alpaca: A strong, replicable instruction-following model. *Stanford Center for Research on Foundation Models. https://crfm. stanford. edu/2023/03/13/alpaca. html*, 3(6):7.

Rohan Taori, Ishaan Gulrajani, Tianyi Zhang, Yann Dubois, Xuechen Li, Carlos Guestrin, Percy Liang, and Tatsunori B. Hashimoto. 2023b. Stanford alpaca: An instruction-following llama model. https://github.com/tatsu-lab/stanford_alpaca.

Ross Taylor, Marcin Kardas, Guillem Cucurull, Thomas Scialom, Anthony Hartshorn, Elvis Saravia, Andrew Poulton, Viktor Kerkez, and Robert Stojnic. 2022. Galactica: A large language model for science. *arXiv preprint arXiv:2211.09085*.

Hugo Touvron, Thibaut Lavril, Gautier Izacard, Xavier Martinet, Marie-Anne Lachaux, Timothée Lacroix, Baptiste Rozière, Naman Goyal, Eric Hambro, Faisal Azhar, et al. 2023. Llama: Open and efficient foundation language models. *arXiv preprint arXiv:2302.13971*.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. *Advances in neural information processing systems*, 30.

Yizhong Wang, Swaroop Mishra, Pegah Alipoormolabashi, Yeganeh Kordi, Amirreza Mirzaei, Atharva Naik, Arjun Ashok, Arut Selvan Dhanasekaran, Anjana Arunkumar, David Stap, Eshaan Pathak, Giannis Karamanolakis, Haizhi Lai, Ishan Purohit, Ishani Mondal, Jacob Anderson, Kirby Kuznia, Krima Doshi, Kuntal Kumar Pal, Maitreya Patel, Mehrad Moradshahi, Mihir Parmar, Mirali Purohit, Neeraj Varshney, Phani Rohitha Kaza, Pulkit Verma, Ravsehaj Singh Puri, Rushang Karia, Savan Doshi, Shailaja Keyur Sampat, Siddhartha Mishra, Sujan Reddy A, Sumanta Patro, Tanay Dixit, and Xudong Shen. 2022. Super-NaturalInstructions: Generalization via declarative instructions on 1600+ NLP tasks. In *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing*, pages 5085–5109, Abu Dhabi, United Arab Emirates. Association for Computational Linguistics.

Jason Wei, Maarten Bosma, Vincent Y. Zhao, Kelvin Guu, Adams Wei Yu, Brian Lester, Nan Du, Andrew M. Dai, and Quoc V. Le. 2022. Finetuned language models are zero-shot learners.

Kichang Yang, Wonjun Jang, and Won Ik Cho. 2022. Apeach: Attacking pejorative expressions with analysis on crowd-generated hate speech evaluation datasets.

Yinfei Yang, Yuan Zhang, Chris Tar, and Jason Baldridge. 2019. PAWS-X: A Cross-lingual Adversarial Dataset for Paraphrase Identification. In *Proc. of EMNLP*.

Lianmin Zheng, Wei-Lin Chiang, Ying Sheng, Siyuan Zhuang, Zhanghao Wu, Yonghao Zhuang, Zi Lin, Zhuohan Li, Dacheng Li, Eric. P Xing, Hao Zhang, Joseph E. Gonzalez, and Ion Stoica. 2023. Judging llm-as-a-judge with mt-bench and chatbot arena.

Chunting Zhou, Pengfei Liu, Puxin Xu, Srini Iyer, Jiao Sun, Yuning Mao, Xuezhe Ma, Avia Efrat, Ping Yu, Lili Yu, Susan Zhang, Gargi Ghosh, Mike Lewis, Luke Zettlemoyer, and Omer Levy. 2023. Lima: Less is more for alignment.

A. Analysis of Performance Across Different Templates and Few-Shot Settings

The evaluation of the models trained with *KIT-19* across a range of benchmarks provides insightful observations into their performance, particularly concerning the type of evaluation templates and few-shot settings. This section delves into the differential impacts of template types and the number of shots on model performance. These analyses are encapsulated in four critical evaluations, as highlighted in Tables 3 to 6.

A.1. Performance in Seen Datasets with the Original Template

Table 3 presents the performance of models evaluated against Seen datasets using the original template. A discernible pattern emerges, indicating that consistency in the template type used during instruction tuning and evaluation positively impacts model performance. Specifically, models trained with our *KIT-19* dataset, KIT-1.3b and KIT-5.8b, demonstrate superiority across all three Seen datasets under this evaluation condition. Moreover, as the number of shots (n=shot) increases, a nuanced performance trend is observed, suggesting that the models benefit variably from the additional contextual information provided by increased shots. This finding underscores the adaptability of models trained with *KIT-19* in leveraging varying amounts of provided context to enhance their performance.

A.2. Performance in Unseen Datasets with the Original Template

The evaluation of models against Unseen datasets with the original template, detailed in Table 4, sheds light on the models' generalization capabilities. Notably, the KIT-1.3b and KIT-5.8b models, despite being evaluated on tasks not seen during the training phase, maintain their performance edge, particularly under specific few-shot settings. This robust performance across unseen tasks highlights the comprehensive nature of the *KIT-19* dataset in preparing models for a broader range of tasks beyond those explicitly encountered during instruction tuning.

A.3. Performance in Seen Datasets with the Custom Template

Table 5 focuses on evaluating models against Seen datasets but pivots to using the custom template. The consistent outperformance of KIT-models underscores the significance of template consistency between training and evaluation phases. The KIT models exhibit remarkable performance improvements, which can be attributed to the tailored instruction delivery mechanism ingrained through the instruction tuning with *KIT-19*. This approach not only enhances model performance on familiar tasks but also exemplifies the models' ability to adhere to varied instructional formats.

A.4. Performance in Unseen Datasets with the Custom Template

Finally, Table 6 illustrates the models' performance against Unseen datasets, evaluated with the custom template akin to the training phase's template structure. Here, the adaptability of the KIT models is once again highlighted, showing a clear performance advantage in most settings. This demonstrates the models' capability to generalize learned instructions to novel tasks effectively, particularly when the evaluation aligns closely with the training template format. The nuanced examination of zero-shot and few-shot settings further reveals the complex interplay between model training and its application versatility in unseen domains.

		KOBEST_COPA (n=shot)				KOBEST_BoolQ (n=shot)				KOBEST_HellaSwag (n=shot)			
Model	**Params**	n=0	n=1	n=2	n=3	n=0	n=1	n=2	n=3	n=0	n=1	n=2	n=3
Polyglot-ko-1.3b	1.3B	71.96%	71.06%	71.06%	72.96%	35.52%	51.27%	49.55%	51.55%	40.13%	38.67%	41.14%	40.35%
Polyglot-ko-5.8b	5.8B	77.55%	75.56%	77.26%	76.75%	43.56%	54.06%	56.47%	56.97%	48.53%	48.47%	48.85%	49.88%
KoAlpaca-5.8b	5.8B	69.77%	68.25%	67.67%	68.25%	50.64%	51.64%	54.70%	53.93%	38.15%	39.24%	38.87%	38.44%
kullm-polyglot-5.8b-v2	5.8B	76.53%	75.87%	76.98%	75.48%	33.71%	53.73%	52.71%	51.72%	44.25%	46.17%	47.15%	47.47%
KIT-1.3b	1.3B	84.09%	83.99%	83.99%	84.18%	33.71%	51.27%	50.17%	48.58%	82.11%	80.51%	78.34%	77.91%
KIT-5.8b	5.8B	**91.38%**	**92.39%**	**93.19%**	**93.69%**	**64.28%**	**61.23%**	**64.72%**	**65.23%**	**98.00%**	**97.80%**	**97.79%**	**97.60%**

Table 3: Experiment results in Seen datasets with the original template: We evaluate 7 models on 6 different benchmark datasets.

		KLUE-STS (n=shot)				KOBEST_SentiNeg (n=shot)				KLUE_YNAT (n=shot)			
Model	**Params**	n=0	n=1	n=2	n=3	n=0	n=1	n=2	n=3	n=0	n=1	n=2	n=3
Polyglot-ko-1.3b	1.3B	59.54%	44.87%	48.08%	49.60%	**68.19%**	54.16%	65.32%	64.97%	33.24%	25.02%	26.01%	28.15%
Polyglot-ko-5.8b	5.8B	**60.34%**	45.30%	39.83%	45.92%	33.95%	66.47%	75.41%	79.49%	33.62%	23.67%	26.76%	29.85%
KoAlpaca-5.8b	5.8B	33.86%	45.25%	46.12%	53.38%	38.48%	61.59%	75.75%	**84.11%**	20.91%	19.15%	19.25%	19.13%
kullm-polyglot-5.8b-v2	5.8B	59.54%	**49.03%**	46.67%	52.42%	33.50%	**74.64%**	68.09%	67.45%	32.20%	21.06%	26.26%	27.28%
KIT-1.3b	1.3B	59.78%	44.87%	39.63%	42.42%	46.23%	65.23%	62.14%	65.38%	**35.46%**	**30.14%**	**29.80%**	**31.69%**
KIT-5.8b	5.8B	58.04%	44.97%	**50.09%**	**57.29%**	42.63%	68.17%	**77.68%**	78.73%	30.89%	24.02%	26.89%	28.42%

Table 4: Experiment results in Unseen datasets with the original template

		KOBEST_COPA (n=shot)				KOBEST_BoolQ (n=shot)				KOBEST_HellaSwag (n=shot)			
Model	**Params**	n=0	n=1	n=2	n=3	n=0	n=1	n=2	n=3	n=0	n=1	n=2	n=3
Polyglot-ko-1.3b	1.3B	67.54%	66.54%	67.06%	66.54%	33.40%	40.53%	43.31%	42.10%	38.40%	38.94%	40.20%	39.74%
Polyglot-ko-5.8b	5.8B	70.74%	69.15%	69.30%	69.99%	55.93%	54.17%	56.38%	54.49%	43.02%	44.36%	45.28%	44.71%
KoAlpaca-5.8b	5.8B	65.40%	66.55%	66.42%	66.41%	39.86%	50.11%	50.92%	53.66%	37.91%	36.97%	36.16%	37.14%
kullm-polyglot-5.8b-v2	5.8B	70.54%	71.85%	72.87%	71.65%	39.84%	55.42%	54.27%	53.18%	41.36%	42.73%	41.85%	44.43%
KIT-1.3b	1.3B	83.78%	82.49%	81.68%	80.88%	34.78%	49.83%	48.34%	46.41%	81.49%	78.95%	76.94%	76.35%
KIT-5.8b	5.8B	**91.59%**	**90.09%**	**90.99%**	**91.79%**	**66.14%**	**60.74%**	**58.27%**	**59.72%**	**97.61%**	**96.80%**	**96.79%**	**96.37%**

Table 5: Experiment results in Seen datasets with the custom template

		KLUE-STS (n=shot)				KOBEST_SentiNeg (n=shot)				KLUE-TC (n=shot)			
Model	**Params**	n=0	n=1	n=2	n=3	n=0	n=1	n=2	n=3	n=0	n=1	n=2	n=3
Polyglot-ko-1.3b	1.3B	59.54%	44.87%	45.30%	33.60%	34.06%	39.37%	50.61%	53.35%	36.86%	22.35%	19.36%	21.23%
Polyglot-ko-5.8b	5.8B	57.90%	44.87%	34.11%	35.10%	33.50%	34.06%	36.03%	38.31%	**45.50%**	15.75%	21.09%	21.38%
KoAlpaca-5.8b	5.8B	58.11%	44.87%	27.68%	28.82%	33.50%	38.90%	51.86%	55.20%	30.64%	22.41%	25.62%	25.13%
kullm-polyglot-5.8b-v2	5.8B	59.34%	44.35%	35.18%	37.00%	33.50%	54.07%	49.03%	46.94%	38.86%	18.62%	19.88%	21.24%
KIT-1.3b	1.3B	56.52%	44.87%	26.70%	28.48%	**80.86%**	**66.33%**	77.55%	79.58%	38.34%	**28.05%**	**29.83%**	**32.00%**
KIT-5.8b	5.8B	**69.71%**	**46.51%**	**45.52%**	**46.29%**	68.98%	34.50%	33.50%	33.50%	28.15%	15.49%	15.02%	14.20%

Table 6: Experiment results in Unseen datasets with the custom template