Solving Korea Dialect Translation Problem Under Data Scarcity

Korea University COSE461 Final Project

Ji Soo Lee
Department of Computer Science
Team 10
2019170989

Hyun Jin Lee
Department of Computer Science
Team 10
2020320023

Jeong Min Moon
Department of Computer Science
Team 10
2020320064

Bo Geon Park
Department of Computer Science
Team 10
2020320134

Abstract

This study develops a translation model to convert Jeju dialect into standard Korean, addressing the challenge of preserving minority languages with limited linguistic data. Traditional neural machine translation (NMT) systems require extensive datasets which are scarce for dialects like Jeju. We adapted the KE-T5 model, originally trained on standard Korean, to include multiple dialects, enhancing its ability to process and generate the Jeju dialect accurately. Additionally, we integrated Reinforcement Learning with Human Feedback (RLHF) to improve the naturalness of translations, addressing the limitations of traditional NMT metrics such as BLEU scores. Our results demonstrate that the multi-dialect training approach significantly enhances model performance, with the application of RLHF promising to increase fluency. This research advances machine translation techniques by effectively integrating multi-dialect training and reinforcement learning, demonstrating significant improvements in translation accuracy and naturalness.[1]

1 Introduction

The preservation and revitalization of minority languages are critical yet challenging endeavors in the field of linguistics. This is exemplified by the Jeju dialect, a unique linguistic form spoken on Jeju Island, South Korea. The Jeju dialect is not only a variant of Korean but is also considered by some linguists to be a distinct language within the Korean language family called Jejueo. The uniqueness and academic value of the Jeju dialect are evident, though it is currently classified by UNESCO as "definitely endangered"[1].

Despite its significance, efforts to preserve the Jeju dialect face substantial hurdles, primarily due to the scarcity of linguistic data available for robust model training. Traditional neural machine translation (NMT) systems, such as those based on the Transformer architecture like the T5 model, rely heavily on large datasets for effective training[2]. The limited dataset size for Jeju dialect presents a significant challenge, as these models require substantial data to capture the nuances of language effectively.

[1]Our code if available at `https://github.com/Ji-Soo-Lee/NJP-Natural-Jeju-Processor`

In response to these challenges, our research adapts proven approaches from other languages to the context of the Jeju dialect, focusing on translation and preservation. Drawing inspiration from methodologies like the Japanese Multi-Dialect Neural Machine Translation (NMT), we employ a multi-dialect training strategy. This approach leverages the shared lexical and syntactic features among multiple dialects to compensate for the lack of extensive single-dialect data[3]. By training the KE-T5 model—originally designed for multifaceted linguistic tasks including language modeling and deshuffling —[4] on multiple dialects, we aim to enhance its understanding and generation of the Jeju dialect.

Furthermore, recognizing the limitations of traditional NMT in maintaining the naturalness and quality of translated text, especially with complex or colloquial expressions, we incorporate Reinforcement Learning with Human Feedback (RLHF)[5]. This technique refines the translation output to better align with human preferences, thus addressing the qualitative aspects that standard BLEU scores may not fully capture.

Our results are promising. The application of a multi-dialect approach, coupled with the fine-tuning of the KE-T5 model and subsequent reinforcement learning adjustments, leads to significant improvements. Notably, the BLEU score, a common metric for evaluating the quality of machine-translated text, shows marked enhancement compared to models trained on single-dialect datasets.

By applying a strategy inspired by successful multilingual models to the Korean context and integrating advanced reinforcement learning techniques, our study not only advances the field of machine translation but also contributes significantly to the preservation efforts of the Jeju dialect. We believe that these methodologies can be adapted for other endangered languages, potentially transforming linguistic preservation efforts globally.

2 Related Work

T5 The T5 model[2], based on the standard vanilla encoder-decoder transformer[6] architecture, represents a unified approach to NLP by converting all text-based language tasks into a text-to-text format. Utilizing techniques like language modeling, a BERT-style objective of masking and predicting words, and input deshuffling[7], T5 serves as a robust framework for transfer learning. This method allows the adaptation of a pre-trained model, heavily reliant on extensive datasets, to effectively manage downstream tasks even with limited data, making it ideal for varied NLP applications.

However, T5's initial training in English poses challenges for applications in other languages like Korean, which differ significantly in syntax and semantics. This gap necessitates a translation of input data from Korean to English, introducing potential errors and inaccuracies, especially when dealing with dialects. This necessitates further adaptations of the T5 model to accommodate the structural nuances of Korean directly, avoiding the pitfalls of cross-linguistic translation errors.

KE-T5 In this context, we adopted the KE-T5[4] model trained with pure Korean by the Korea Electronics Technology Institute Artificial Intelligence Research Center (KETI). Trained with the massive dataset of Korean, KE-T5 can more richly understand Korean structure and knowledge from it. Furthermore, since the Jeju dialect has a similar structure as the standard form of Korean, we can directly use the dialect form as an input to fine-tune the model.

Nonetheless, there still exists a limit to simply using the KE-T5 model. Even though fine-tuning the model is not training the model from the bottom, it requires a sufficient amount of data and resources to obtain desired output in time. Unfortunately, most dialectal forms of languages have too little data to fine-tune the KE-T5 model effectively. Here we've come to search for alternative approaches.

Japanese Multi-Dialect NMT One Japanese team has presented Neural Machine Translation (NMT) that translates multiple Japanese dialects into standard forms[3]. Following same method as what google had done with multilingual NMT[8], they used the same stacked LSTM encoder and decoder model. Interestingly, instead of training NMT with a multilingual dataset, they tried it on a multi-dialect of Japanese and got higher performance than Statistical Machine Translation, believed to be much more accurate compared to NMT.

One issue for building Machine Translation with Japanese dialect was lack of training data. To overcome this limitation, they have chosen to concatenate multi-dialect as a hole training dataset. Since google has shown probabilities of concatenating language datasets[8], it seems reasonable to sum up multi-dialect because dialects rooting from one language likely to have more similarities than collected different languages. Therefore, we wanted to try this method to our model.

Since their multi-dialect model used stacked LSTM for encoder and decoder with simple attention, it had weakness in arranging structure compared to self-attention of the transformer. To generate sentences in the structure that we aim to, Neural Machine should be more flexible and in that sense we chose T5 model for our project as it is more advanced for fine-tuning in general purposes.

Cost Effective RLHF To enhance the translation capabilities of our model, we leverage Reinforcement Learning with Human Feedback (RLHF) and use human evaluations to directly influence the training process. This forces the model to optimize translations based on human preferences rather than just matching the statistical patterns in the data. However, modeling translation preferences for reinforcement learning requires high quality preference data, and obtaining such data is costly even if it is assisted by generative AI like ChatGPT. We leverage the induction bias of 'high-quality human translation is superior to machine-generated translation' to collect preference data at a lower cost[5]. By training the reward model to contrast between machine translations with high-quality human translations, we can utilize the reward model to furnish the feedback to the language model.

3 Approach

3.1 KE-T5-Small

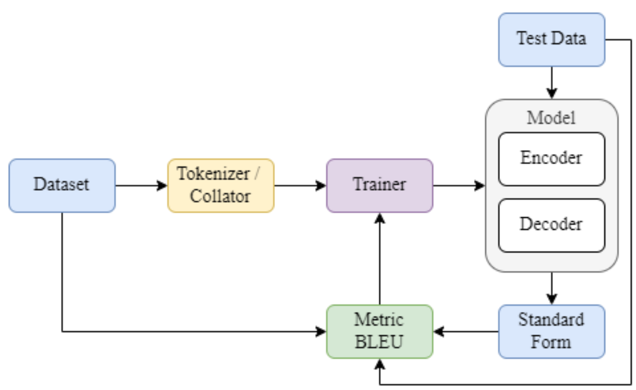

Figure 1: Single-dialect Training Process

Figure 1 explains how we trained our single jeju-dialect model and tested its performance. For our research, we selected the KE-T5[4] Small model as our baseline due to its compatibility with the text-to-text paradigm of the T5 architecture and its resource efficiency. T5's unified framework treats all NLP tasks as text generation problems, simplifying the model architecture and enabling high performance across diverse tasks. The KE-T5 Small model, a Korean version of the T5 architecture, leverages the strengths of T5 and is also trained in Korean, making it well-suited to our specific problem context.

We chose the Small variant to address practical constraints related to computational resources and time. The Small model requires less memory and computational power, making it feasible for limited hardware and quicker to train. This choice balances strong performance with the realities of limited resources and time, ensuring our research is both effective and feasible.

We followed a fine-tuning methodology[9] using the KE-T5-Small, a pre-trained model checkpoint available on Hugging Face[10], trained for both Korean and English based on the T5 architecture. The fine-tuning process is slightly modified to adapt the model to our specific translation tasks between Korean and the Jeju dialect.

We loaded the pre-trained KE-T5-Small model checkpoint along with essential components such as the Tokenizer, Collator, Seq2Seq Model, and Seq2Seq Trainer using transformer modules. These components were critical in processing our preprocessed dataset of Jeju dialect and corresponding Korean standard form, formatted to meet the model's requirements.

For evaluation, we utilized the BLEU score to assess the accuracy of the translated text against a set of reference translations. This metric allowed us to quantitatively refine the model's performance during the fine-tuning process, optimizing it for higher accuracy in translating between Korean and the Jeju dialect.

3.2 Multi-dialect

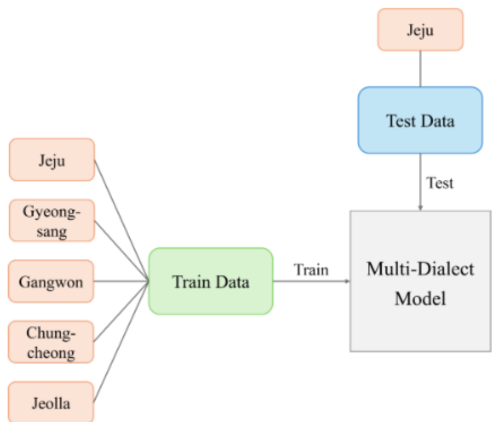

Figure 2: Multi-dialect Training Process

Figure 2 explains how we trained our multi-dialect model and tested its performance.

We utilized the previously described KE-T5 small model, and trained it using a multi-dialect dataset as training data. The proportion of each dialect in the dataset was set to be equal, and their order was randomized to prevent any influence on the training process.

Once the training was completed, the Jeju dialect dataset was used as test data to evaluate the performance of the model. Through this process, we could verify whether the findings of Abe et al. (2018)[3] can be applied to the Korean Jeju dialect. In other words, when it is difficult to obtain enough Jeju dialect data, we could expect performance improvements by using other Korean dialects as a part of train data.

3.3 RLHF

We applied RLHF to our base T5 model and multi-dialect T5 model in the hopes of improving the translation quality. The traditional RLHF method requires a large high-quality dataset of human evaluations to train the reward model. It is costly and very labor-intensive. Instead, we optimized our reward model γ to capture translation preferences by contrasting the differences between high-quality human translation from our dataset and machine translation from our model:

$$\mathcal{L}(r) = -\mathbb{E}_{(x,y_w,y_l)\sim D_{rm}} \left[\log \sigma \left(r(x, y_w) - r(x, y_l)\right)\right] \tag{1}$$

x represents the source language sentence, y_w and y_l denote a high-quality human translation and a machine-generated translation, and $D_{rm} = \{\left(x^{(i)}, y_w^{(i)}, y_l^{(i)}\right)\}_{i=1,\ldots,N}$ is the preference dataset[5].

Figure 3 explains how our RL model works. Our T5 models are fine-tuned using PPO algorithm based on the reward signal from our reward model. The reward model outputs high rewards for translations that are fluent and similar to the ones human generated. We hope to enhance the translation capabilities of our models and generate human-like translations as the PPO algorithm trains our T5 models in a stable manner based on the rewards.

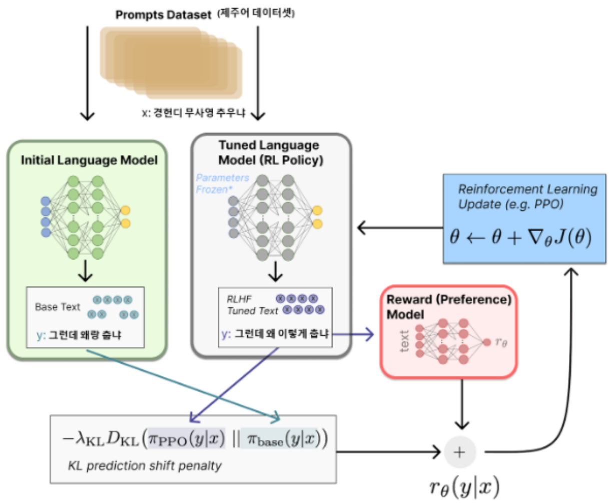

Figure 3: Reinforcement Learning Process

4 Experiments

4.1 Dataset

We used "The utterance of a Korean dialect(한국어 방언 발화)" dataset from AI Hub[11]. It consists of two parts, voice data and text data. Voice data consists of voice conversations of people who speak local dialects, and text data is a text-type version of it. We preprocessed this text data to obtain standard form and dialect form of each sentence. The dataset includes dialects from various provinces of Korea, consisting of Jeju, Chungcheong, Gyeongsang, Gangwon, and Jeolla. We also used "Korean dialect data for middle and old people(중 · 노년층 한국어 방언 데이터)" dataset. The speakers in the dataset consisted of people over 50 years old. We divided these datasets into training data and test data at a ratio of 11:1.

4.2 Evaluation Method

BLEU Score We used BiLingual Evaluation Understudy(BLEU) score to evaluate the performance of our model. BLEU score is a widely used metric for machine translation performance evaluation. It measures how similar a sentence generated by a translator is to a reference sentence. However, BLEU score indeed has limitations due to its emphasis on precision without directly considering recall. This focus on precision means that BLEU can sometimes miss important aspects of how much of the reference translation is captured by the machine translation.

METEOR Score A well-known alternative metric we also used to address this limitation is METEOR (Metric for Evaluation of Translation with Explicit ORdering). METEOR calculates the harmonic mean of precision and recall, effectively balancing these two aspects. It evaluates word-to-word matches between the translated and reference texts, adjusting for stemming and synonymy, and incorporates a measure of sentence structure alignment. This results in a more nuanced evaluation of translation quality, especially in capturing both the accuracy of the translation (precision) and the completeness (recall).

4.3 Experimental Detail

Single-dialect KE-T5 Small Approximately 350,000 Jeju dialect sentences were used as a dataset which is divided into 325,000 sentences for training and 25,000 sentences for validation while 10,000 sentences were used separately for the test. The learning rate was set to 5e-4, weight decay was set to 0.01, and the train batch size was set to 32. Our model was trained with a training dataset and a validation dataset and subsequently, we generated sample sentences for comparison and test using test dataset. The training time took about 40 minutes with the L4 GPU.

Multi-dialect KE-T5 Small A total of 525,000 dialect sentences were used as a dataset which is sliced into a train and valid set with a similar ratio as before and 10,000 of the test set is used separately. Not affecting the test set, about 180,000 lines of sentences were sampled from each local dialect dataset proportion of 45,000 per each dialect, and the dataset was finally constructed by randomly mixing the order of them to perform fine-grained training. Same as before, the learning rate was set to 5e-4, weight decay was set to 0.01, and the train batch size was set to 32. The training time took about an hour with the L4 GPU.

Reward Model Over 300,000 pairs of human translations and machine generated translations were used to train the reward model. The reward model is trained by fine-tuning "KLUE-BERT" model on sequence classification task. The model is optimized to distinguish human translation from our models' translations. The initial learning rate for AdamW optimizer is set to 5e-5 and the batch size is set to 32. The training time took about 3 hours with L4 GPU.

Fine-tuning via PPO A total of 50,000 Jeju dialect sentences were used to fine-tune our translation models during RLHF training. Our translation models (base model and multi-dialect model) are fine-tuned using PPO algorithm[12] with our reward model. The initial learning rate for AdamW optimizer is set to 1.41e-5 and the batch size is set to 64. The training time took about 3.5 hours with T4 GPU.

4.4 Results

Dataset	BLEU	METEOR
Single Jeju Dialect	68.26	0.83
Multi-Dialect	**73.38**	**0.86**
Single Jeju + RL	66.27	0.81
Multi-Dialect + RL	72.33	0.85
GPT4o	23.20	0.37

Table 1: BLEU and METEOR scores for our methods and GPT4o for reference.

Impact of Using Multi-Dialect Data Both the BLEU and METEOR scores are higher for the Multi-Dialect dataset compared to the Single Jeju Dialect dataset. This improvement likely stems from the increased diversity and richness of the training data. By including multiple dialects, the model can learn a broader range of linguistic structures and vocabulary, leading to translations that are more robust and generally closer to the reference translations. This aligns with expectations as more comprehensive training data typically yields better model performance.

Effect of Reinforcement Learning (RL) Interestingly, the addition of RL to both Single Jeju Dialect and Multi-Dialect datasets results in a slight decrease in both BLEU and METEOR scores. While one might initially expect that RL would enhance performance by optimizing specific aspects of the translation process towards more natural language usage, it appears to have reduced the exactness of word-to-word or phrase-to-phrase matches as measured by these metrics. This could be because RL potentially focuses on enhancing aspects such as fluency and naturalness, which might not always align with the exact phrases used in the reference texts.

Limitations of BLEU and METEOR Scores The fundamental nature of BLEU and METEOR as metrics focuses heavily on the similarity between the machine-generated text and a set of reference texts. While these metrics are useful for quantitatively assessing certain dimensions of translation quality (like precision and recall), they may not fully capture the qualitative aspects of translation, such as fluency and idiomatic correctness, which are often enhanced through methods like RL. Thus, a decrease in these scores does not necessarily indicate a worse translator, but perhaps one that is prioritizing different aspects of language generation.

5 Analysis

5.1 Using Multi-dialect Data

Original	[‘몸이 자꾸 고료운 거 보난 니가 신 셍이여’]
Reference	[‘몸이 자꾸 **가려운** 거 보니 **이가 있는** 모양이야’]
Single T5	[‘몸이 자꾸 **고료운** 거 보니 **이 있는** 모양이야’]
Multi T5	[‘몸이 자꾸 **가려운** 것을 보니 **이가 있는** 모양이야’]
Single T5 BLEU score	19.13
Multi T5 BLEU score	50.00

Table 2: Comparative translations showing the effectiveness of multi-dialect training.

Table 2 above shows that the model trained by a single Jeju dialect dataset didn't translate the original Jeju dialect form sentence into a complete standard form that fits the grammar. For example, ‘니가 신’ in original sentence should be translated as ‘이가 있는’, but the particle ‘가’ wasn't generated well, resulting in ‘이 있는’. Also, ‘고료운’ should be translated as ‘가려운’, but it didn't work out. On the other hand, the model trained by a multi-dialect dataset, the result was close to the reference sentence. It generated ‘이가 있는’ and ‘가려운’ very well. From these examples, we could see that the model could learn diverse linguistic characteristics of the Korean language, as described in the work of Abe et al. (2018)[3].

5.2 Using Reinforcement Learning

Original	[‘집 앞이 손 매는 건 집에 아픈 사름이 시나 굿을 허나 다 이유가 싯주게’]
Reference	[‘집 앞에 손 매는 **건** 집에 아픈 사람이 있거나 굿을 하거나 다 이유가 있지’]
RLHF Model	[‘집 앞에 손 매는 **것은** 집에 아픈 사람이 있거나 굿을 하거나 다 이유가 있지’]
Multi T5	[‘집 앞에 손 매는 **건** 집에 아픈 사람이 있거나 굿을 하거나 다 이유가 있지’]
RLHF BLEU score	78.25
Multi T5 BLEU score	100.00

Table 3: Comparative translations showing the impact of RLHF on translation quality.

Original	[‘우리 두린 땐 하르방들이 초신 삼아 주민 그거 신어낫주게’]
Reference	[‘우리 **어렸을** 땐 할아버지들이 짚신 삼아 주면 그거 신었었지’]
RLHF Model	[‘우리 **어릴** 땐 할아버지들이 짚신 삼아 주면 그거 신었었지’]
Multi T5	[‘우리 **어렸을** 땐 할아버지들이 짚신 삼아 주면 그거 신었었지’]
RLHF BLEU score	75.06
Multi T5 BLEU score	100.00

Table 4: Comparative translations showing the impact of RLHF on translation quality.

In Table 3, we can see that our RLHF model is producing well translated sentences, but is receiving lower BLEU scores than our T5 model due to alterations from reference sentences. For example, in the first sentence, “매는 건” is more of a colloquial expression while “매는 것은” is formal expression.

Output	Reward
집 앞에 손 매는 **건** 집에 아픈 사람이 있거나 굿을 하거나 다 이유가 있지	3.49
집 앞에 손 매는 **것은** 집에 아픈 사람이 있거나 굿을 하거나 다 이유가 있지	3.59

Table 5: Output of reward model

Our RLHF model generates sentences in formal expressions as our reward model outputs higher scores for sentences in formal expressions compared to sentences in colloquial expressions, which we can see in Table 5 This may be because our reward model has seen more data in formal expressions than in colloquial expressions during training. Accordingly, we can predict that our RLHF model performs worse on BLEU or METEOR score than T5 models because RLHF models are generating sentences under the influence of the reward model instead of simply matching the statistical patterns and outputting the exact phrases from reference texts.

Original	[‘보린 몰레왓디 농사가 잘 안 돼’]
Reference	[‘**보리는** 모래밭에 농사가 잘 안 돼’]
RLHF Model	[‘**보리는** 모래밭에 농사가 잘 안 돼’]
Multi T5	[‘**보린** 모래밭에 농사가 잘 안 돼’]
RLHF score	100.00
Multi T5 score	75.98

Table 6: Comparison of RLHF and Multi T5 model translations with performance scores.

Original	[‘옛날 수도 엇인 땐 집이 큰일 나민 동네 사름덜이 다 물 질어다 어’]
Reference	[‘옛날 수도 없을 **때는 집에** 큰일 나면 동네 사람들이 다 물 길어다 줬어’]
RLHF Model	[‘옛날 수도 없을 **때는 집에** 큰일 나면 동네 사람들이 다 물 길어다 줬어’]
Multi T5	[‘옛날 수도 없을 **땐 집이** 큰일 나면 동네 사람들이 다 물 길어다 줬어’]
RLHF score	100.00
Multi T5 score	67.03

Table 7: Comparison of RLHF and Multi T5 model translations with performance scores.

From Table 6 and Table 7, we can see that RLHF has still advanced the translation capabilities of our T5 models. If the translation model generates a poor translation, during RLHF training, the reward model penalizes the translation model and optimizes the model to generate a human-like translation, which improves the translation quality.

Based on human evaluations, no significant difference was observed between RLHF models and T5 models in the overall assessment. Although RLHF models demonstrated the ability to produce more accurate translations than T5 models in certain instances, there were also occasions where it generated less accurate translations. This is because the language capability of reward model is crucial for preference learning[5]. However, due to financial and resource constraints, we were unable to develop a sufficiently intricate reward model.

6 Conclusion

This study focuses on converting the Jeju dialect to standard Korean, contributing to advancements in the field of machine translation. Through the development and fine-tuning of the KE-T5 Small model, coupled with multi-dialect training strategies and reinforcement learning, we have made substantial progress in addressing the challenges associated with translating endangered dialects with limited data resources.

Our primary findings confirm the effectiveness of using a multi-dialect approach for training machine translation models when data for a specific dialect is sparse. The multi-dialect model outperformed the single-dialect model, as evidenced by higher BLEU and METEOR scores, demonstrating the utility of leveraging linguistic similarities across various dialects to enhance translation accuracy.

Additionally, we explored the application of RHLF, which, despite causing a slight decrease in BLEU and METEOR scores, showed potential in improving the naturalness and fluency of translations. This suggests that RLHF could be valuable in contexts where the qualitative aspects of language, such as idiomatic correctness, are prioritized.

The primary limitations of our work include its focus on one-way translation, specifically from the Jeju dialect to standard Korean, without accommodating reverse translation capabilities. Additionally, due to financial constraints, we were unable to employ more advanced models such as GPT-4 for reinforcement learning, which could potentially improve the model's performance further.

For future work, we plan to extend our model's capabilities to convert standard Korean back into the Jeju dialect, enhancing its utility. Another avenue could involve utilizing more advanced models such as GPT-4 for RLAIF, aiming to refine the feedback loop and potentially enhance the translation quality beyond what current metrics can capture.

References

[1] Jejueo in Republic of Korea — en.wal.unesco.org. `https://en.wal.unesco.org/countries/republic-korea/languages/jejueo`.

[2] Colin Raffel, Noam Shazeer, Adam Roberts, Katherine Lee, Sharan Narang, Michael Matena, Yanqi Zhou, Wei Li, and Peter J. Liu. Exploring the limits of transfer learning with a unified text-to-text transformer. *Journal of Machine Learning Research*, 21(140):1–67, 2020.

[3] Kaori Abe, Yuichiroh Matsubayashi, Naoaki Okazaki, and Kentaro Inui. Multi-dialect neural machine translation and dialectometry. In Stephen Politzer-Ahles, Yu-Yin Hsu, Chu-Ren Huang, and Yao Yao, editors, *Proceedings of the 32nd Pacific Asia Conference on Language, Information and Computation*, Hong Kong, 1–3 December 2018. Association for Computational Linguistics.

[4] San Kim, Jin Yea Jang, Minyoung Jung, and Saim Shin. A model of cross-lingual knowledge-grounded response generation for open-domain dialogue systems. In Marie-Francine Moens, Xuanjing Huang, Lucia Specia, and Scott Wen-tau Yih, editors, *Findings of the Association for Computational Linguistics: EMNLP 2021*, pages 352–365, Punta Cana, Dominican Republic, November 2021. Association for Computational Linguistics.

[5] Nuo Xu, Jun Zhao, Can Zu, Sixian Li, Lu Chen, Zhihao Zhang, Rui Zheng, Shihan Dou, Wenjuan Qin, Tao Gui, Qi Zhang, and Xuanjing Huang. Advancing translation preference modeling with RLHF: A step towards cost-effective solution. February 2024.

[6] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Ł ukasz Kaiser, and Illia Polosukhin. Attention is all you need. In I. Guyon, U. Von Luxburg, S. Bengio, H. Wallach, R. Fergus, S. Vishwanathan, and R. Garnett, editors, *Advances in Neural Information Processing Systems*, volume 30. Curran Associates, Inc., 2017.

[7] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. BERT: Pre-training of deep bidirectional transformers for language understanding. In Jill Burstein, Christy Doran, and Thamar Solorio, editors, *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 4171–4186, Minneapolis, Minnesota, June 2019. Association for Computational Linguistics.

[8] Melvin Johnson, Mike Schuster, Quoc V. Le, Maxim Krikun, Yonghui Wu, Zhifeng Chen, Nikhil Thorat, Fernanda Viégas, Martin Wattenberg, Greg Corrado, Macduff Hughes, and Jeffrey Dean. Google's multilingual neural machine translation system: Enabling zero-shot translation. *Transactions of the Association for Computational Linguistics*, 5:339–351, 2017.

[9] Jo, Joonu (metamath@gmail.com). ML simple works - A Gentle Introduction to Creating an English-to-Korean translator with Transformers — metamath1.github.io. `https://metamath1.github.io/blog/posts/gentle-t5-trans/gentle_t5_trans.html`.

[10] KETI-AIR/ke-t5-small · Hugging Face — huggingface.co. `https://huggingface.co/KETI-AIR/ke-t5-small`.

[11] AI-Hub — aihub.or.kr. `https://www.aihub.or.kr/`.

[12] PPO Trainer — huggingface.co. `https://huggingface.co/docs/trl/ppo_trainer`.

A Appendix: Reference Code

A.1 KE-T5-Samll Translator

KETI-AIR/ke-t5-small · Hugging Face — huggingface.co. `https://huggingface.co/KETI-AIR/ke-t5-small`.

A.2 RLHF PPO Trainer

PPO Trainer — huggingface.co. `https://huggingface.co/docs/trl/ppo_trainer`.

B Appendix: Team contributions

Ji Soo Lee

Data preprocessing, presentation, and main report writing

Hyun Jin Lee

RLHF implementation, training, and report writing

Jeong Min Moon

T5 model implementation, training, and evaluation, and report writing

Bo Geon Park

Data preprocessing, and main report writing