Informative Evidence-guided Prompt-based Fine-tuning for English-Korean Critical Error Detection

Dahyun Jung[1], **Sugyeong Eo**[1], **Chanjun Park**[2], **Hyeonseok Moon**[1]
Jaehyung Seo[1], **Heuiseok Lim**[1*]
[1]Korea University, South Korea, [2]Upstage, South Korea
{dhaabb55,djtnrud,glee889,seojae777,limhseok}@korea.ac.kr
chanjun.park@upstage.ai

Abstract

Critical error detection (CED) aims to identify the presence of catastrophic meaning distortion in machine translation. Fatal errors require significant attention because of their potential to cause personal or societal harm. The CED for Korean, an agglutinative language, is particularly highlighted, as minor variations in morphemes often bring substantial shifts in semantic interpretation. However, research on Korean is still underexplored and has room for improvement. In this study, we conduct the first investigation of CED for English–Korean to the best of our knowledge. We adopt prompt-based fine-tuning and propose various informative evidence to incorporate into the input prompt. Subsequently, we perform comprehensive verification and analysis to identify the most helpful guidance for detecting critical errors. The experimental results show that prompt-based fine-tuning with informative evidence outperforms standard fine-tuning by a large margin, demonstrating its remarkable effectiveness in English–Korean CED.[1]

1 Introduction

The remarkable progress of neural machine translation (NMT) has considerably facilitated an exchange of information and communication among speakers from various countries, lowering barriers to global communication. Particularly, services leveraging large-scale language models (LLMs) have been increasingly integrated into various real-world scenarios, spanning daily lives and business industries (Brown et al., 2020; Lin et al., 2021; Zhang et al., 2022; Scao et al., 2022).

Despite the increasing advancements in machine translation (MT) technology, MT systems still struggle with translation errors, as they fail to consider the context of sentences or cultural differences in languages (Bender et al., 2021). Users who lack proficiency in the target language may find it difficult to discern the types and extent of errors in the translation output. A quality estimation (QE) task addresses this issue to provide users with feedback regarding the reliability of an MT output. QE predicts translation quality by referencing only the source and MT sentence, without the necessity for human reference translation. This offers a significant advantage in real-world scenarios where the reference translations do not exist (Specia et al., 2009, 2020).

Furthermore, critical error detection (CED) emerged as a sub-task of QE at 2021 Sixth Conference on Machine Translation (WMT21) (Specia et al., 2021), particularly focusing on detecting cases where translation errors result in fatal distortions in meaning (Raunak et al., 2022; Zerva et al., 2022). The importance of CED lies in its ability to detect errors that may incur severe consequences. For example, cases where the meaning distortion in the translation output could be perceived as offensive or have the potential to cause social, legal, or economic harm. Although the occurrence of critical errors in MT results is a long-tail problem, preventing even one translation error from incurring a serious social issue is crucial (Martindale et al., 2021). With the advent of LLMs such as GPT–3 (Brown et al., 2020) and ChatGPT (OpenAI-Blog, 2022), users are increasingly relying on MT for various purposes. Therefore, ensuring the reliability of the translation output generated by these models is vital, and CED is an essential component in the verification process.

CED is a binary classification task, consisting of five error types: *toxicity*, *safety*, *named entity*, *number*, and *sentiment*. The error types are commonly applied across languages, but the detectable error range can be changed based on the characteristics of each language. Therefore, it is necessary to define types that reflect language-specific properties. The recently released English–Korean CED dataset

*Corresponding Author
[1]The code is available at https://github.com/ekgus9/PBFT-for-KoCED.

Proceedings of the 13th International Joint Conference on Natural Language Processing and the 3rd Conference of the Asia-Pacific Chapter of the Association for Computational Linguistics (Volume 1: Long Papers), pages 344–358
November 1–4, 2023.

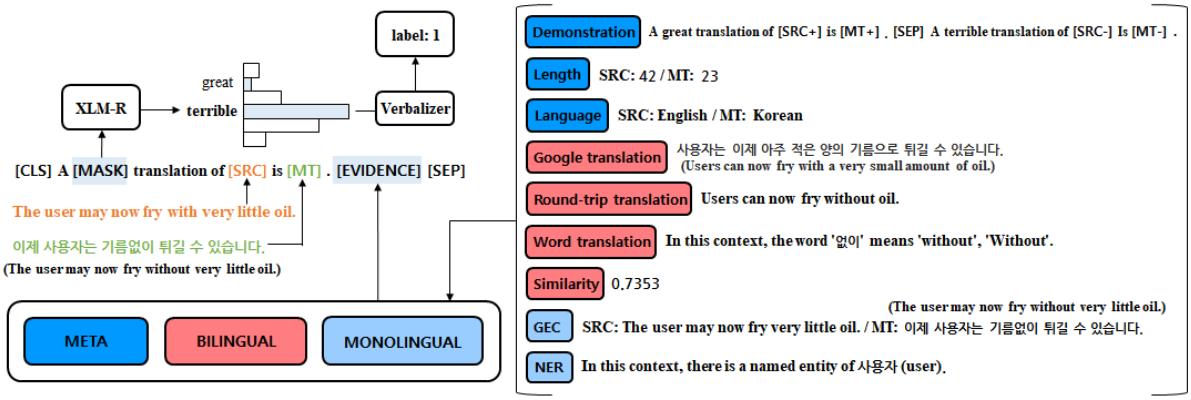

Figure 1: Overall architecture of prompt-based fine-tuning with informational evidence

additionally introduces a *politeness* label reflecting the cultural property of Korea. *Politeness* tags capture cases where incorrect honorific expressions are used, which can be considered impolite behavior in certain situations. Thus, the model trained with a culture-aware English–Korean CED dataset can well filter translation errors that LLMs may overlook in terms of courtesy in Korea.

However, the English–Korean CED still remains underexplored. In Korean, an agglutinative language, the meaning changes sensitively based on a few variations of morphemes. Namely, even a character-level change would immediately distort the meaning of the source to a completely different one. This highlights the importance of detecting catastrophic errors in English–Korean translation. In this study, we perform experiments using the English–Korean CED dataset to facilitate the inspection of critical errors in the language pair. To the best of our knowledge, this is the first study that experiments with the task. Although fine-tuning the model with pre-trained LLM is the *de-facto* standard for adapting to downstream tasks, we focus on the gap between the learning objectives of pre-training and fine-tuning (Liu et al., 2023; Schick and Schütze, 2021; Liu et al., 2021a). We adopt prompt-based fine-tuning in our experiments to reduce this gap and maximize the language understanding capability of LLM. As illustrated in Figure 1, this study also explores more extensive informative evidence that can be combined as input for error detection, classified into META, BILINGUAL, and MONOLINGUAL evidence. With the experiments using informative evidence and various combinations of templates and verbalizers, we report the results and analysis of which evidence contributes to the performance improvement with which templates. We demonstrate the effectiveness of prompt-based learning and informative evidence in CED through extensive experiments using various prompts and evidence. Our main contributions are the following:

- We perform prompt-based fine-tuning for the English–Korean CED task, which has not been sufficiently explored in previous research.
- We propose multiple informative evidence that can be helpful in detecting critical errors in MT and report the most efficient evidence with analysis.
- Our experimental results outperform the fine-tuning performance by a significant margin, demonstrating the effectiveness of prompt-based methods with informative evidence in CED.

2 Related Works

2.1 Critical Error Detection (CED)

CED task aims to determine the existence of a critical error in a translation at the sentence level, using binary labels (Zerva et al., 2022). The occurrence of the error can cause miscommunication, potentially resulting in grave consequences (Sharou and Specia, 2022; Sudoh et al., 2021). For instance, mistranslations of crucial medical information directly affect the lives of patients. This semantic distortion manifests in several patterns and has diverse schema types, details of which are expounded upon in Appendix A.

Various language pairs are leveraged in CED task to address the critical error issue. Rubino et al.

(2021); Jiang et al. (2021); Chen et al. (2021) perform CED on English–German, English–Chinese, English–Czech, and English–Japanese pairs. Rubino et al. (2021) train a model by leveraging a large amount of synthetic data produced using parallel corpora and MT systems. Meanwhile, Jiang et al. (2021) explore weighted sampling to deal with imbalanced datasets and refine the architecture by extracting sentence features. Chen et al. (2021) propose learning method of using a pre-trained model and task-specific classifier. Eo et al. (2022) present prompt-based fine-tuning with demonstrations and Google MT on English–German and Portuguese–English.

As observed in previous studies, the investigation regarding critical error has been diversely explored. However, no studies have been conducted using English–Korean CED datasets. We use the corresponding dataset in this study as the data additionally introduces the language-specific properties.

2.2 Prompt-based Fine-tuning

Prompt-based fine-tuning has been proposed to address these challenges, offering a method that reformulates tasks to more effectively leverage pre-trained knowledge (Liu et al., 2023; Schick and Schütze, 2021; Liu et al., 2021a; Brown et al., 2020).

Schick and Schütze (2021) introduce pattern-exploiting training (PET), which combines reformulating tasks as cloze styles with a fine-tuning approach. Schick and Schütze (2020b) combine PET and ALBERT to achieve good performance with a considerably smaller number of parameters. Gao et al. (2020) propose a model to automatically generate prompts and demonstrations for prompt-based fine-tuning approaches, and Liu et al. (2021a) employ a method to automatically search for discrete prompts in continuous space.

To the best of our knowledge, no prior research has executed prompt-based learning utilizing suitable techniques to identify the distortion of meaning in MT. We propose a method to perform English–Korean CED using prompt-based fine-tuning, which has previously demonstrated impressive results across numerous fields. Particularly, we generate prompts fitting the task by including evidence relevant to English-to-Korean translation and evaluate the influence of these prompts on the model's performance.

3 Proposed Methods

3.1 Preliminary

In this task, we derive the CED model to predict error label y given source sentence x_{src} and translation sentence x_{mt}. Specifically, we apply prompt-based learning that bridges the gap between the pre-training and fine-tuning (Gao et al., 2020). We adopt a pre-trained language model XLM-RoBERTa (Conneau et al., 2019) that is trained with masked language modeling (MLM) objectives. Considering these, we construct a template that reformulates the CED task as a cloze style that aims to fill the masked part in the given input text (Petroni et al., 2019; Cui et al., 2021).

More detailed procedures are shown below. We denote x_{prompt} as a form of input that incorporates a template containing `[MASK]` token, x_{src} and x_{tgt}. For instance, in executing a binary critical error detection task, one designed example for our prompt as follows:

$$x_{prompt} = \texttt{[CLS]}\ \text{A}\ \texttt{[MASK]}\ \text{translation of}\ x_{src}\ \text{is}\ x_{mt}\ .\ \texttt{[SEP]}$$

where the prompt varies based on the template. For the given x_{prompt}, the CED model is supervised to predict the appropriate word in the `[MASK]` position, such as "great" (non-error) or "terrible" (error).

Furthermore, we define $\psi : y \in Y \rightarrow w \in W$ as a function called the verbalizer that maps the label $y \in Y$ to the label word $w_y \in W_Y$ (*i.e.* $\psi(y) = w_y$). In this case, Y denotes the label set of a targeting task (*e.g.* $Y = \{\text{NOT}, \text{ERR}\}$), and W_Y denotes the corresponding set of label words (*e.g.* $W_Y = \{\text{great}, \text{terrible}\}$). We formulate the probability of predicting class $y \in Y$ as

$$P(y|x_{src}, x_{mt}) = \frac{P(\texttt{[MASK]} = w_y \mid x_{prompt})}{\sum_{w \in W_Y} P(\texttt{[MASK]} = w \mid x_{prompt})} \quad (1)$$

In our study, we aim to find the most effective task-specific prompt x_{prompt} and verbalizer function ψ. To enhance explainability, we perform manual template engineering that follows human intuition (Brown et al., 2020; Schick and Schütze, 2020a).

3.2 Evidence Investigation

Providing informative evidence as part of the model's input acts as supplementary descriptions

in detecting critical errors. We discover factors that directly or indirectly impact model training by incorporating various information during prompt-based fine-tuning. In order to capture and assess the diverse facets of MT, we categorize informative evidence into three categories: (1) META evidence: additional information that is not directly related to the translation task but can provide valuable insights and guidance, (2) BILINGUAL evidence: additional information obtained through the comparison and alignment of source and MT results, (3) MONOLINGUAL evidence: additional information that utilizes only one of the source or MT sentences. The categories are described in detail as follows.

3.2.1 Leveraging Meta Data as Evidence

In-context learning with demonstration In-context learning is a method that provides explanations or examples in the input context, enabling the model to inform the task should perform (Brown et al., 2020). We employ the approach to enhance the model's comprehension of the CED task via demonstrations. We utilize a positive and negative labeled sample to be included in the prompt input to clarify critical errors (*e.g.* "A `[label word+]` translation of `[SRC+]` is `[MT+]` .`[SEP]` A `[label word-]` translation of `[SRC-]` Is `[MT-]` ."). The demonstration sample should be randomized according to labels.

Explicit language identification When predicting labels, we use information about the language of the source sentence or translation sentence. This additional task aims to capture the benefits of explicitly specifying language identification. mBART (Liu et al., 2020), which has learned multiple languages, uses the strategy of including explicit information about these languages. Therefore, the utilization of this evidence presupposes that the comprehension of sentences can be enhanced by discerning the language in which each sentence is written. We create a prompt template by referring to the recommended prompt guidance from the OpenAI playground for the default sentence-level translation task[2].

Reducing hallucination error To identify the hallucination error present in sentences, we make use of the information regarding the length of either the source or the target sentence. This approach draws inspiration from Berard et al. (2019); Guerreiro et al. (2022), where they use length-based filtering when building an MT system and remove a substantial number of hallucination errors. If a concise translation sentence is generated for a long source sentence, it may indicate that the MT model has omitted or misunderstood necessary information. Therefore, we hypothesize that checking length information can be beneficial in detecting MT errors.

3.2.2 Leveraging Bilingual Data as Evidence

Integration of results from commercialized MT system We use the Google MT system to translate the source sentence and utilize the result as the supplementary description. Previous studies have found that leveraging the translated result of a commercialized MT system improves the MT-related task's performances (Chen et al., 2021; Wang et al., 2020; Moon et al., 2021). We speculate that it would also be meaningful to provide Google MT results to detect errors in MT.

Extracting restored semantics from source We apply round-trip translation using the commercial system to detect critical errors. Round-trip translation refers to translating a translated sentence back to the source language, enabling us to see how well the MT system translated through the source (Somers, 2005). If there is a translation error in the target, there will also exist an error in the sentence translated back to the source′ through round-trip translation. Therefore, we better understand the errors made by the MT system by comparing the original source and the source′ created through round-trip translation.

Fine-grained phrase-level control We induce a model to better understand the translation of the entire sentence by providing word-by-word translation results. We provide the model with the translated results of words included in the source or translation sentence. This approach suggests using bilingual dictionary information in the CED task (Ghazvininejad et al., 2023). If the word exists in the dictionary, we translate the meaning of the word and compose it into the prompt. For example, if the word 'apple' is included in the dictionary, the prompt would be constructed as "the word 'apple' means '사과'." This improves translation accuracy and supports users in obtaining more natural translation results. Such helpful hints in translation can also be useful in detecting errors in translation tasks.

Measuring the quality score We present the similarity between the source and target sentences as additional evidence. In MT, the higher the sim-

[2] https://platform.openai.com/docs/guides/completion

ilarity between the two sentences, the better the translation quality considered (Chan and Ng, 2008). Therefore, if there is an error in the translation, it will also affect the similarity between the two sentences. For example, if the sentence "I am going to the Mcdonald's" is incorrectly translated to "I am going to the Mcdoanld's," the similarity between the words "Mcdonald's" and "Mcdoanld's" decreases. Such sentence similarity assists in quality evaluation and error identification.

3.2.3 Leveraging Monolingual data as Evidence

Grammatical denoising We identify and rectify grammatical errors in the source sentences and their translations, and use the corrected sentences during training. Our hypothesis posits that if the model learns from sentences containing grammatical errors, it may encounter confusion when discerning critical errors. Grammatical error correction (GEC) results eliminate confusion caused by grammatical errors, enabling more accurate critical error detection.

Mitigating named entity error We can better discern named entity errors that occur in the MT system by providing the Named entity recognition (NER) results of each sentence. NER is a task that recognizes entities related to a specific person, location, organization, etc., in a sentence. NER assists the model to recognize detailed semantic tokens that improve translation quality (Liu et al., 2021b). Thus, it draws understanding in NMT by utilizing the syntactic and semantic structure of natural language. For example, in the sentence "John works at Apple," it is crucial to recognize the named entities "John" and "Apple." However, there may be cases where the model fails to recognize "Apple" and misinterprets it as fruit "apple". We allow the model to identify named entity errors by providing NER results in each sentence.

3.3 Evidence Mix-up

Through ensemble, more accurate predictions are secured by integrating the prediction results of each model. We train separate models using the informative evidence and aim to improve performance by aggregating these model outputs. First, to evaluate relative effectiveness by evidence category, we conduct experiments for each category. Also, we select the top-k outputs trained on each informative evidence based on the Matthews correlation coefficient (MCC) to observe the performance variation with the number of models used. Then, we use majority voting to combine the results from different prompts (Lester et al., 2021; Hambardzumyan et al., 2021). This method is based on the principle of majority rule, selecting the most frequently chosen result among the predictions made by each model as the final prediction. We expect that high performance is achieved by integrating the prediction results from various models.

Label	Category	Train	Dev	Test
NOT		6,606	444	924
ERR	TOX	133	6	7
	SAF	122	15	10
	NAM	95	12	20
	NUM	116	6	12
	SEN	110	12	14
	POL	83	5	13
Total		7,265	500	1,000

Table 1: Statistics of an English–Korean CED Dataset

4 Experiments and Results

This section evaluates prompt-based fine-tuning with informative evidence using the English–Korean CED dataset.

4.1 Dataset Details

We consider the English–Korean critical error detection (KoCED) dataset[3], which consists of the source sentence, translation sentence, critical error presence/absence label, and detailed error tag. The dataset is constructed utilizing parallel corpora of daily life Korean-English text. Unlike domain-specific datasets, the sentences in this daily life corpus span various fields, including travel, food, retail, and real estate, offering diverse expressions and a broad vocabulary.

There are six detailed errors: *toxicity*, *safety*, *named entity*, *number*, *sentiment*, and *politeness*. If multiple errors appear in a single sentence, one type with the most critical is annotated. Statistical information regarding the dataset is presented in Table 1.

[3] https://aihub.or.kr/aihubdata/data/view.do?dataSetSn=71269

	Test set				Eval set				Mean
Method	**MCC**	**F1-NOT**	**F1-ERR**	**F1-Multi**	**MCC**	**F1-NOT**	**F1-ERR**	**F1-Multi**	
mBERT	0.0030	0.9550	0.0227	0.0217	0.2061	0.9411	0.1791	0.1685	0.3122
XLM-R-base	0.2588	0.9565	0.2807	0.2685	0.3567	0.9458	0.3590	0.3395	0.4707
XLM-R-large	0.4307	0.9661	0.4286	0.4140	0.6346	0.9648	0.6444	0.6218	0.6381
PBFT	0.6564	0.9770	0.6667	0.6513	0.7208	0.9710	0.7451	0.7235	0.7640
+META									
Demo	0.6396	0.9759	0.6512	0.6355	**0.7836**	**0.9779**	**0.7872**	**0.7699**	0.7775
Language	0.5931	0.9739	0.5950	0.5795	0.6731	0.9681	0.6813	0.6596	0.7155
Length	0.6050	0.9751	0.5913	0.5766	0.7230	0.9726	0.7191	0.6994	0.7328
+BILINGUAL									
GMT	**0.6649**	**0.9770**	**0.6815**	**0.6658**	0.7331	0.9721	0.7573	0.7362	0.7735
RTT	0.6539	0.9770	0.6614	0.6462	0.7646	0.9755	0.7843	0.7651	**0.7785**
WT	0.6417	0.9765	0.6452	0.6300	0.6616	0.9670	0.6739	0.6516	0.7309
Similarity	0.6155	0.9756	0.6034	0.5887	0.6506	0.9658	0.6667	0.6439	0.7137
+MONOLINGUAL									
GEC	0.5523	0.9725	0.5357	0.5210	0.7230	0.9726	0.7191	0.6994	0.7120
NER	0.5345	0.9707	0.5378	0.5221	0.6616	0.9670	0.6739	0.6516	0.6899

Table 2: The main result of the models for English–Korean Critical Error Detection. This table shows the experimental outcomes for the test and evaluation dataset. Bold indicates the best performance. We use the XLM-RoBERTa-large model, which performed excellently in the baseline, in both prompt-based fine-tuning and prompt-based fine-tuning with additional information. **PBFT** is prompt-based fine-tuning, and in PBFT training, additional resources categorized as META, BILINGUAL, and MONOLINGUAL evidence are presented with a +; **Demo** is the value experimented with on the model that supplied the demonstration; **GMT** utilizes Google Translate system; **RTT** is round-trip translation that translates the translation sentence back to the source language; **WT** indicates the provision of guidance for translations on a word-by-word basis; **Mean** is the result of averaging all the scores for each method.

4.2 Training Details

Models For training, all models are implemented with PyTorch[4] and Transformers[5]. We utilize the pre-trained language models 'bert-base-multilingual-cased', 'xlm-roberta-base', and 'xlm-roberta-large' checkpoints. We use a batch size of 64, the Adam optimizer with a learning rate of 2e–5, and train for 10 epochs. The experiments are performed on an NVIDIA RTX A6000 environment.

Baselines We compare our proposed methods with the standard fine-tuning approach. We set the fine-tuning results of the multilingual BERT (mBERT) (Pires et al., 2019) and XLM-RoBERTa (Conneau et al., 2019) models as a baseline. mBERT is a multilingual model based on BERT, trained on Wikipedia data in 104 different languages, not limited to English. XLM-RoBERTa is a multilingual version of RoBERTa (Liu et al., 2019), a pre-trained model in over 100 languages.

Evidences To determine token length, we employ the XLM-RoBERTa-large tokenizer. We use the Google MT API[6] as a commercial translation system. Also, we utilize Sentence-BERT (Reimers and Gurevych, 2019) to calculate the similarity score between the source and translation sentences. The model is obtained from hugging-face using the 'distiluse-base-multilingual-cased-v1', which includes Korean. GEC and NER data are obtained through the 'gec' and 'ner' tasks of the Pororo platform[7].

4.3 Evaluation Details

We automatically measure MCC and F1 scores as our evaluation metrics to evaluate the performance of the model. MCC is a metric used in binary classification tasks. This value can be meaningfully utilized to measure the accuracy of the model. F1 is a metric calculated as the harmonic mean of precision and recall. Precision represents the propor-

[4] https://pytorch.org/
[5] https://huggingface.co/
[6] https://translate.google.com/
[7] https://github.com/kakaobrain/pororo

Method	MCC	F1-NOT	F1-ERR	F1-Multi
META	0.6330	0.9621	0.6667	0.6414
BILINGUAL	**0.6928**	**0.9798**	**0.6833**	**0.6700**
MONOLINGUAL	0.5903	0.9740	0.5882	0.5729
All	0.6732	0.9788	0.6552	0.6413
Top-7	0.6829	0.9793	0.6667	0.6529
Top-5	0.6833	0.9793	0.6723	0.6583
Top-3	0.6833	0.9793	0.6723	0.6583

Table 3: The ensemble result of prompt models with informational evidence. We report performance on the test dataset. We denote ALL as the models that contain all informational evidence. Top-k is the result of selecting and ensembling k-models with the highest performance.

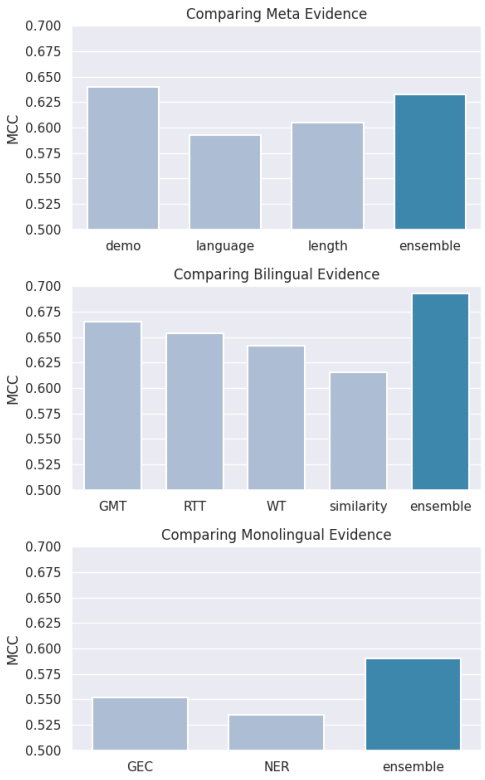

Figure 2: Evidence and Mix-up modeling performance of each category

tion of samples predicted as positive by the model that are actually positive, and recall represents the proportion of actual positive samples that are predicted as positive by the model. F1 score, which considers both these metrics, provides a more accurate assessment of the model's performance. The scoring is based on the code utilized in WMT 21[8].

4.4 Experimental Results

4.4.1 Main Results

We present the experimental results of comparing prompt-based fine-tuning with informative evidence in Table 2. We initially perform a baseline experiment to identify the optimal model for the CED. We find that XLM-RoBERTa-large is the most effective model, which we further leverage it in our subsequent work.

Our proposed method of experimenting with prompt-based fine-tuning significantly outperforms fine-tuning strategy by a large margin. The performance of models leveraging BILINGUAL information demonstrates a substantial improvement compared to the baseline. Adding Google MT outperforms the XLM-RoBERTa-Large baseline by 0.2342 on MCC and 0.2518 on F1-Multi in our test set, the best of all evidence. It also outperforms prompt-based fine-tuning without any evidential support. By incorporating BILINGUAL information, the models can effectively incorporate both source and target sentences, enabling a synergistic interaction between them. Our experimental findings corroborate the advantageous nature of emphasizing the interaction between these two components in translation-related tasks. The inclusion of META information, such as demonstrations, exhibits promising performance, indicating its utility in enhancing the reasoning process essential for prompt approaches. In addition, We conclude that the enhancements come from the use of prompts, which effectively bridge the gap between pre-training and downstream tasks (Gao et al., 2020). In fact, our strategy enables us to optimize the utilization of an off-the-shelf LM by implementing a simple and effective prompt scheme.

The other approach, utilizing MONOLINGUAL information, demonstrates comparable results relative to the baseline. However, it falls short of surpassing the advantage achieved by standard prompt-based fine-tuning. This observation suggests that relying solely on single language information in MT potentially introduce noise and hinder overall performance.

4.4.2 Empirical Prompt Engineering

We present the results of the experiments in this section in Appendix B due to page limitations. We experiment with different templates and verbalizers for all prompt-based ways to find the appropriate prompts for our task. To minimize performance variation due to prompt engineer-

[8] https://github.com/sheffieldnlp/qe-eval-scripts/tree/master/wmt21

ing, we conduct at least four examinations per method. First, we attempt to find the appropriate prompt for CED using standard prompt-based fine-tuning. Prompts formatted as natural sentences are more task-robust compared to unnaturally formatted prompts. 'A `[MASK]` translation of `[SRC]` is `[MT]`.' or '`[SRC]` `[MT]` is `[MASK]`.' yield better results. Given this observation, we perform training with additional features based on task-appropriate and natural templates.

A straightforward transition to a more efficient prompt, in the absence of any additional techniques, is capable of producing significant outcomes. Particularly, in the prompt-based fine-tuning method of Table 6, the prompt with the highest performance exhibits a difference of 0.195 compared to the lowest performance. This proves that finding task-specific prompts is essential for prompt-based fine-tuning. The result allows us to observe the advantages of manual prompt engineering, which is that it enables the creation of prompts that are more aligned with human intuition.

Among the examined features, Google MT and round-trip translation exhibit the highest performance, showcasing their effectiveness in considering both source and target languages. Furthermore, these methods demonstrate robustness in handling prompt changes. The minimum score achieved by prompts with Google MT is overwhelmingly higher than the maximum score achieved by prompts with NER. The obtained results follow our expectation that the additional information is effective in identifying catastrophic errors regardless of the prompt selection.

4.4.3 Modeling with a Mix of Informative Evidence

Utilizing the informative evidence from the previous section, we perform experiments with prompt ensembling to identify the right combination of evidence. Table 3 is the result of ensembling the models with prompt-based fine-tuning. The employment of ensemble techniques usually leads to enhanced results. MCC score of BILINGUAL ensemble increases by 0.0279 compared to prompt-based fine-tuning with Google MT, which is the highest score of the individual models. This demonstrates the effectiveness of creating diverse models using informative evidence and then ensembling them.

The results indicate that using all informative evidence is ineffective in achieving optimal performance. This is because the model that includes the top-k pieces of information performs better than the model that includes all the information. In particular, the improvement observed through model filtering suggests the importance of selecting meaningful models when constructing an ensemble model. Nevertheless, as evidenced by the BILINGUAL ensemble performance, it is crucial to ensure the inclusion of relevant and meaningful information in the task, rather than solely relying on high performance or an abundance of information (Figure 2).

Method	MCC	F1-NOT	F1-ERR	F1-Multi
PBFT+BILINGUAL	**0.6928**	**0.9798**	**0.6833**	**0.6700**
PBFT	0.6564	0.9770	0.6667	0.6513
- 10%	0.5479	0.9700	0.5616	0.5449
- 20%	0.4954	0.9684	0.5013	0.4855
- 30%	0.5796	0.9725	0.5858	0.5698

Table 4: Performance with data distribution. The percentage value is the extent to which the number of error-labeled data was reduced to balance the data. We present the scores obtained through prompted fine-tuning of the sampled data, and evaluate these values by comparing them to the performance of an ensemble model incorporating BILINGUAL evidence, which emerges as the most successful among our approaches.

4.4.4 Impact of Data Distribution

The dataset we use is skewed with a high percentage of error labels. To investigate the impact of data balancing on efficiency, we perform an experiment using a method that balances the data. The data sampling process involves averaging the results from three random data samplings. Table 4 verifies that data sampling is generally less efficient compared to prompt-based fine-tuning using the entire dataset. These results suggest that the influence of label imbalance in the task is insignificant, indicating that our evidence addition methodology holds more significance than attempting to achieve uniformity across the data.

5 Conclusion

In this study, we presented a prompt-based fine-tuning approach with informational evidence in an English–Korean CED. We analyzed the impact of prompt engineering on CED and conducted experiments using the engineering approach that yields the best performance. Also, we explore informational evidence to support our prompt-based ap-

proach and apply it to our learning. Our prompt-based fine-tuning with informative evidence outperformed standard fine-tuning and prompt-based fine-tuning. Among the evidence, the model that considers both source and target languages yielded the best performance, and we achieved the best results using templates composed of natural, fluent sentences. While LLMs have recently demonstrated a remarkable ability to generate natural language, our study contributes to reducing the unintended social impact by preventing and filtering out the catastrophic results they produce.

Limitations

Our experiment is restricted to the English–Korean language pair, and its extension to other languages is not directly examined. It remains to be seen whether our method exhibits universal applicability across all languages, or if modifications are required for languages with distinct structural or linguistic attributes. In addition, the computational cost and time associated with prompt and answer engineering may not be practical for application. The process requires significant human intervention for the design, selection, and assessment of corresponding prompts and evidences. Future research should aim to address these limitations, and we plan to explore these to validate the robustness and applicability of our empirical observations.

Ethics Statement

Our methodology employs an automated MT system facilitated by Google MT. This system exhibits certain biases, for instance, in the context of gender, wherein gender-neutral nouns are rendered as gender-specific in the target languages (Prates et al., 2020). Besides, the PLMs included in this study are mBERT and XLM-RoBERTa models. The mBERT is pre-trained on the top 104 languages with the largest Wikipedia. The XLM-RoBERTa is pre-trained on 2.5TB of filtered CommonCrawl data containing 100 languages. It is important to note that all our disseminated models may potentially bear biases inherited from these source corpora.

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). This research was supported by the MSIT(Ministry of Science and ICT), Korea, under the ITRC(Information Technology Research Center) support program(IITP-2023-2018-0-01405) supervised by the IITP(Institute for Information & Communications Technology Planning & Evaluation). This work was supported by Institute for Information & communications Technology Planning & Evaluation(IITP) grant funded by the Korea government(MSIT) (No. 2022-0-00369, (Part 4) Development of AI Technology to support Expert Decision-making that can Explain the Reasons/Grounds for Judgment Results based on Expert Knowledge).

References

Emily M Bender, Timnit Gebru, Angelina McMillan-Major, and Shmargaret Shmitchell. 2021. On the dangers of stochastic parrots: Can language models be too big?. In *Proceedings of the 2021 ACM conference on fairness, accountability, and transparency*, pages 610–623.

Alexandre Berard, Ioan Calapodescu, and Claude Roux. 2019. Naver labs europe's systems for the wmt19 machine translation robustness task. *arXiv preprint arXiv:1907.06488*.

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.

Yee Seng Chan and Hwee Tou Ng. 2008. MAXSIM: A maximum similarity metric for machine translation evaluation. In *Proceedings of ACL-08: HLT*, pages 55–62, Columbus, Ohio. Association for Computational Linguistics.

Yimeng Chen, Chang Su, Yingtao Zhang, Yuxia Wang, Xiang Geng, Hao Yang, Shimin Tao, Guo Jiaxin, Wang Minghan, Min Zhang, Yujia Liu, and Shujian Huang. 2021. HW-TSC's participation at WMT 2021 quality estimation shared task. In *Proceedings of the Sixth Conference on Machine Translation*, pages 890–896, Online. Association for Computational Linguistics.

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*.

Leyang Cui, Yu Wu, Jian Liu, Sen Yang, and Yue Zhang. 2021. Template-based named entity recognition using BART. In *Findings of the Association for Computational Linguistics: ACL-IJCNLP 2021*, pages

1835–1845, Online. Association for Computational Linguistics.

Sugyeong Eo, Chanjun Park, Hyeonseok Moon, Jaehyung Seo, and Heuiseok Lim. 2022. KU X upstage's submission for the WMT22 quality estimation: Critical error detection shared task. In *Proceedings of the Seventh Conference on Machine Translation (WMT)*, pages 606–614, Abu Dhabi, United Arab Emirates (Hybrid). Association for Computational Linguistics.

Tianyu Gao, Adam Fisch, and Danqi Chen. 2020. Making pre-trained language models better few-shot learners. *arXiv preprint arXiv:2012.15723*.

Marjan Ghazvininejad, Hila Gonen, and Luke Zettlemoyer. 2023. Dictionary-based phrase-level prompting of large language models for machine translation. *arXiv preprint arXiv:2302.07856*.

Nuno M Guerreiro, Pierre Colombo, Pablo Piantanida, and André FT Martins. 2022. Optimal transport for unsupervised hallucination detection in neural machine translation. *arXiv preprint arXiv:2212.09631*.

Karen Hambardzumyan, Hrant Khachatrian, and Jonathan May. 2021. Warp: Word-level adversarial reprogramming. *arXiv preprint arXiv:2101.00121*.

Genze Jiang, Zhenhao Li, and Lucia Specia. 2021. ICL's submission to the WMT21 critical error detection shared task. In *Proceedings of the Sixth Conference on Machine Translation*, pages 928–934, Online. Association for Computational Linguistics.

Brian Lester, Rami Al-Rfou, and Noah Constant. 2021. The power of scale for parameter-efficient prompt tuning. *arXiv preprint arXiv:2104.08691*.

Xi Victoria Lin, Todor Mihaylov, Mikel Artetxe, Tianlu Wang, Shuohui Chen, Daniel Simig, Myle Ott, Naman Goyal, Shruti Bhosale, Jingfei Du, Ramakanth Pasunuru, Sam Shleifer, Punit Singh Koura, Vishrav Chaudhary, Brian O'Horo, Jeff Wang, Luke Zettlemoyer, Zornitsa Kozareva, Mona T. Diab, Veselin Stoyanov, and Xian Li. 2021. Few-shot learning with multilingual language models. *CoRR*, abs/2112.10668.

Pengfei Liu, Weizhe Yuan, Jinlan Fu, Zhengbao Jiang, Hiroaki Hayashi, and Graham Neubig. 2023. Pre-train, prompt, and predict: A systematic survey of prompting methods in natural language processing. *ACM Computing Surveys*, 55(9):1–35.

Xiao Liu, Yanan Zheng, Zhengxiao Du, Ming Ding, Yujie Qian, Zhilin Yang, and Jie Tang. 2021a. Gpt understands, too. *arXiv preprint arXiv:2103.10385*.

Ye Liu, Yao Wan, Jian-Guo Zhang, Wenting Zhao, and Philip S Yu. 2021b. Enriching non-autoregressive transformer with syntactic and semanticstructures for neural machine translation. *arXiv preprint arXiv:2101.08942*.

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.

Yinhan Liu, Myle Ott, Naman Goyal, Jingfei Du, Mandar Joshi, Danqi Chen, Omer Levy, Mike Lewis, Luke Zettlemoyer, and Veselin Stoyanov. 2019. Roberta: A robustly optimized bert pretraining approach. *arXiv preprint arXiv:1907.11692*.

Marianna Martindale, Kevin Duh, and Marine Carpuat. 2021. Machine translation believability. In *Proceedings of the First Workshop on Bridging Human–Computer Interaction and Natural Language Processing*, pages 88–95.

Hyeonseok Moon, Chanjun Park, Sugyeong Eo, Jaehyung Seo, and Heuiseok Lim. 2021. An empirical study on automatic post editing for neural machine translation. *IEEE Access*, 9:123754–123763.

OpenAI-Blog. 2022. Chatgpt: Optimizing language models for dialogue.

Fabio Petroni, Tim Rocktäschel, Sebastian Riedel, Patrick Lewis, Anton Bakhtin, Yuxiang Wu, and Alexander Miller. 2019. Language models as knowledge bases? In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 2463–2473, Hong Kong, China. Association for Computational Linguistics.

Telmo Pires, Eva Schlinger, and Dan Garrette. 2019. How multilingual is multilingual bert? *arXiv preprint arXiv:1906.01502*.

Marcelo OR Prates, Pedro H Avelar, and Luís C Lamb. 2020. Assessing gender bias in machine translation: a case study with google translate. *Neural Computing and Applications*, 32:6363–6381.

Vikas Raunak, Matt Post, and Arul Menezes. 2022. Salted: A framework for salient long-tail translation error detection. *arXiv preprint arXiv:2205.09988*.

Nils Reimers and Iryna Gurevych. 2019. Sentence-bert: Sentence embeddings using siamese bert-networks. *arXiv preprint arXiv:1908.10084*.

Raphael Rubino, Atsushi Fujita, and Benjamin Marie. 2021. NICT Kyoto submission for the WMT'21 quality estimation task: Multimetric multilingual pretraining for critical error detection. In *Proceedings of the Sixth Conference on Machine Translation*, pages 941–947, Online. Association for Computational Linguistics.

Teven Le Scao, Angela Fan, Christopher Akiki, Ellie Pavlick, Suzana Ilić, Daniel Hesslow, Roman Castagné, Alexandra Sasha Luccioni, François Yvon,

Matthias Gallé, et al. 2022. Bloom: A 176b-parameter open-access multilingual language model. *arXiv preprint arXiv:2211.05100*.

Timo Schick and Hinrich Schütze. 2020a. Few-shot text generation with pattern-exploiting training. *arXiv preprint arXiv:2012.11926*.

Timo Schick and Hinrich Schütze. 2020b. It's not just size that matters: Small language models are also few-shot learners. *arXiv preprint arXiv:2009.07118*.

Timo Schick and Hinrich Schütze. 2021. Exploiting cloze-questions for few-shot text classification and natural language inference. In *Proceedings of the 16th Conference of the European Chapter of the Association for Computational Linguistics: Main Volume*, pages 255–269, Online. Association for Computational Linguistics.

Khetam Al Sharou and Lucia Specia. 2022. A taxonomy and study of critical errors in machine translation. In *Proceedings of the 23rd Annual Conference of the European Association for Machine Translation*, pages 171–180, Ghent, Belgium. European Association for Machine Translation.

Harold Somers. 2005. Round-trip translation: What is it good for? In *Proceedings of the Australasian Language Technology Workshop 2005*, pages 127–133.

Lucia Specia, Frédéric Blain, Marina Fomicheva, Erick Fonseca, Vishrav Chaudhary, Francisco Guzmán, and André F. T. Martins. 2020. Findings of the WMT 2020 shared task on quality estimation. In *Proceedings of the Fifth Conference on Machine Translation*, pages 743–764, Online. Association for Computational Linguistics.

Lucia Specia, Frédéric Blain, Marina Fomicheva, Chrysoula Zerva, Zhenhao Li, Vishrav Chaudhary, and André F. T. Martins. 2021. Findings of the WMT 2021 shared task on quality estimation. In *Proceedings of the Sixth Conference on Machine Translation*, pages 684–725, Online. Association for Computational Linguistics.

Lucia Specia, Marco Turchi, Nicola Cancedda, Nello Cristianini, and Marc Dymetman. 2009. Estimating the sentence-level quality of machine translation systems. In *Proceedings of the 13th Annual conference of the European Association for Machine Translation*, Barcelona, Spain. European Association for Machine Translation.

Katsuhito Sudoh, Kosuke Takahashi, and Satoshi Nakamura. 2021. Is this translation error critical?: Classification-based human and automatic machine translation evaluation focusing on critical errors. In *Proceedings of the Workshop on Human Evaluation of NLP Systems (HumEval)*, pages 46–55, Online. Association for Computational Linguistics.

Minghan Wang, Hao Yang, Hengchao Shang, Daimeng Wei, Jiaxin Guo, Lizhi Lei, Ying Qin, Shimin Tao, Shiliang Sun, Yimeng Chen, and Liangyou Li. 2020. HW-TSC's participation at WMT 2020 quality estimation shared task. In *Proceedings of the Fifth Conference on Machine Translation*, pages 1056–1061, Online. Association for Computational Linguistics.

Chrysoula Zerva, Frédéric Blain, Ricardo Rei, Piyawat Lertvittayakumjorn, José G. C. de Souza, Steffen Eger, Diptesh Kanojia, Duarte Alves, Constantin Orăsan, Marina Fomicheva, André F. T. Martins, and Lucia Specia. 2022. Findings of the WMT 2022 shared task on quality estimation. In *Proceedings of the Seventh Conference on Machine Translation (WMT)*, pages 69–99, Abu Dhabi, United Arab Emirates (Hybrid). Association for Computational Linguistics.

Susan Zhang, Stephen Roller, Naman Goyal, Mikel Artetxe, Moya Chen, Shuohui Chen, Christopher Dewan, Mona Diab, Xian Li, Xi Victoria Lin, et al. 2022. Opt: Open pre-trained transformer language models. *arXiv preprint arXiv:2205.01068*.

	Input (source sentence)	Translation sentence with critical error	Translation sentence without critical error
TOX	The person who was with the cat last night is black.	어젯밤 고양이를 괴롭힌 사람은 흑인입니다. The person who harassed the cat last night is black.	어젯밤 고양이와 함께 있던 사람은 흑인입니다. The person who was with the cat last night is black.
SAF	Touching electronic devices, power plugs, and outlets with wet hands is prohibited as they should not be exposed to water.	전자 기기, 전원 플러그, 콘센트는 물에 노출되지 않아야 하므로 젖은 손으로 만지는 것은 괜찮습니다. Touching electronic devices, power plugs, and outlets with wet hands is fine as they should not be exposed to water.	전자 기기, 전원 플러그, 콘센트는 물에 노출되지 않아야 하므로 젖은 손으로 만지는 것은 금지되어 있습니다. Touching electronic devices, power plugs, and outlets with wet hands is prohibited as they should not be exposed to water.
NAM	The main aspect of touring the Grand Canyon is the view.	그랜드 캔 여행의 주요 측면은 전망입니다. The main aspect of touring the Grand Can is the view.	그랜드 캐년 여행의 주요 측면은 전망입니다. The main aspect of touring the Grand Canyon is the view.
NUM	As I mentioned yesterday, I want more than two boxes of ripe bananas.	어제 언급했듯이 잘 익은 바나나 두 상자를 원합니다. As mentioned yesterday, I want two boxes of ripe bananas.	어제 언급했듯이 잘 익은 바나나 두 상자 이상을 원합니다. As I mentioned yesterday, I want more than two boxes of ripe bananas.
SEN	I think your cake looks amazing, but I'm on a diet so I can't have any.	케이크가 정말 끔찍해 보이는데 다이어트 중이라서 못 먹겠어요. The cake looks really awful, but I'm on a diet so I can't have any.	케이크가 정말 맛있어 보이는데 다이어트 중이라서 못 먹겠어요. I think your cake looks amazing, but I'm on a diet so I can't have any.
POL	I would like to express my sincere apologies for becoming upset during the meeting.	회의 중에 화를 내어 진심으로 미안하다. I am truly sorry for becoming upset during the meeting. (rude)	회의 중에 화를 낸 점에 대해 진심으로 사과드립니다. I would like to express my sincere apologies for becoming upset during the meeting. (polite)

Table 5: Critical Error Example by Category. We provide critical error cases of translation classified into six types.

A Pattern and Schema of Critical Error

A.1 Pattern in which the critical error appears

Deviations in meaning from the original sentence can appear as mistranslation, hallucination, or deletion (Zerva et al., 2022). First, mistranslation is an error that occurs when a source sentence is incorrectly translated, resulting in a distortion of its original meaning. If the sentence "I'm feeling blue." is translated as feeling the color blue, it is a mistranslation because the expression indicates feeling sad or depressed in English. Second, hallucination is an error where the content not present in the source sentence is introduced into the translation. If an MT system translates "I'm going to the store" as "I'm going to the store. I'm a freak," it indicates an error where unnecessary content is added during the translation process. Third, deletion is an error where content present in the source sentence is removed from the translation. Essential information is omitted when translating the sentence "I went to the store and bought some apples," as "I went to the store."

A.2 Critical Error Schema

Critical errors encompass universal characteristics, such as *toxicity*, *safety*, *named entity*, *number* and *sentiment*, which are applicable to the majority of languages. In English–Korean CED, *politeness* is considered as a type of error beyond the aforementioned error types.

***Toxicity* (TOX)** is an error type that covers derogatory or offensive content targeting specific race, religion, gender, and force, which can be encountered in translation output. These toxic expressions might originate in the source text or be introduced during the translation process. Particularly, TOX undermines the credibility of the document and leads to ethical issues. For instance, in Table 5, the generated translation maximizes biased opinions about black people, which is a semantic distortion that can potentially result in social discrimination.

***Safety* (SAF)** refers to a type of error that may provoke safety harm due to the wrong translation. For example, an incorrect translation of a product manual may lead to inappropriate usage and potentially

compromise user safety. This issue is of even greater significance in the medical domain, as inaccuracies in translated medical documents may result in severe or even fatal outcomes. The SAF example in Table 5 depicts the potential safety hazards stemming from erroneous translations. If a user follows the translated statement, this could pose a serious and potentially life-threatening risk.

***Named Entity* (NAM)** refers to a type of error where the name of an entity, such as a person, place name, organization, and date, is not properly represented in the translation. Such errors can result in substantial information loss and distortion of the original meaning. As shown in the NAM example of Table 5, "the Grand Canyon" is mistranslated as "the Grand Can," substantially deviating from the intended entity. These inaccuracies can undermine the meaning of the original sentence and impede effective communication.

***Number* (NUM)** refers to an error type associated with the mistranslation of numeric entities, such as times and dates. These errors can have serious consequences, particularly when dealing with sensitive documents. The NUM example in Table 5 represents the potential risk of mistranslating quantities, which may result in a loss of trust. Compromising the integrity of quantities or dates can lead to the degradation of the document's content, causing commercial harm.

***Sentiment* (SEN)** is a type of translation error that changes the polarity of a sentence, thus distorting its meaning or conveying an incorrect sentiment. This type of error is particularly impactful in the marketing domain, where conveying the wrong sentiment can negatively affect the perception of a brand or product. Additionally, if a sentence's sentiment is reversed, as exemplified by the SEN example presented in Table 5, it may convey unintended blame or criticism.

***Politeness* (POL)** denotes an error where the translated statements exhibit contextually inappropriate or impolite expressions. This error type is particularly relevant to specific languages that incorporate politeness within their syntactic structure. Particularly, Korean is characterized by a clear distinction between formal and informal speeches. This distinction amplifies the potential impact of out-of-context issues, as it may inadvertently render a sentence rude or offensive. Table 5 provides an example of an informal Korean translation derived from a polite English expression[9]. In a professional setting, the use of such informal language may upset others.

Identifying these critical error types enables the mitigation of severe issues. Furthermore, it provides valuable feedback to MT systems, highlighting areas in need of refinement.

B Experiments with templates and verbalizers

[9]"미안하다(sorry)" may sound rude in a polite setting. The polite form "죄송합니다(sorry)" have to be used.

Method	Template	Verbalizer	MCC	F1-NOT	F1-ERR	F1-Multi
PBFT	**src mt mask**	great / terrible	0.6125	0.9757	0.5818	0.5677
		good / bad	0.5877	0.9740	0.5812	0.5661
		! / ?	0.5500	0.9725	0.5272	0.5128
		great / error	0.5855	0.9740	0.5739	0.5590
		yes / no	0.6246	0.9737	0.6475	0.6304
	src mt is **mask**	great / terrible	0.6317	0.9760	0.6341	0.6189
		great / error	0.5984	0.9750	0.5932	0.5781
	src mt? **mask**	good / bad	0.5497	0.9719	0.5470	0.5316
		yes / no	0.5984	0.9745	0.5932	0.5781
	src mt. It was **mask** translation.	great / terrible	0.5855	0.9734	0.5901	0.5745
		good / bad	0.5230	0.9716	0.4490	0.4362
	A **mask** translation of **src** is **mt**.	great / terrible	0.6035	0.9751	0.5841	0.5695
		great / error	0.6368	0.9760	0.6457	0.6302
		good / bad	**0.6564**	**0.9770**	**0.6667**	**0.6513**
	source: **src**, transration: **mt** is **mask**	great / terrible	0.4614	0.9671	0.4655	0.4502
	Translate from **src** to **mt**: **mask**	great / terrible	0.5746	0.9735	0.5614	0.5465
		! / ?	0.5531	0.9718	0.5546	0.5390
	src mt. **mask** translation	great / terrible	0.5682	0.9717	0.5827	0.5662
		good / bad	0.5351	0.9714	0.5263	0.5112
		great / error	0.5709	0.9735	0.5455	0.5310
+Demo	**src mt** is **mask demo_ok demo_bad**	great / error	0.6024	0.9751	0.5766	0.5622
	A **mask** translation of **src** is **mt**. **demo_ok demo_bad**	good / bad	0.6089	0.9750	0.6050	0.5899
	demo_ok demo_bad src mt mask	yes / no	0.5855	0.9740	0.5739	0.5590
	demo_ok demo_bad src mt is **mask**	great / error	0.5960	0.9739	0.6016	0.5859
	demo_ok demo_bad A **mask** translation of **src** is **mt**.	good / bad	**0.6396**	**0.9759**	**0.6512**	**0.6355**
+Language	**src mt** is **mask**. **gmt**	great / terrible	0.5711	0.9723	0.5806	0.5646
	english source: **src** korean transtation: **mt** is **mask**	great / terrible	0.4880	0.9687	0.4870	0.4717
		great / error	**0.5931**	**0.9739**	**0.5950**	**0.5795**
	A **mask** translation of [en] **src** is [ko] **mt**.	great / terrible	0.5817	0.9728	0.5920	0.5759
		yes / no	0.5103	0.9662	0.5401	0.5219
	en **src** ko **mt mask**	great / error	0.5614	0.9730	0.5405	0.5259
+Length	**src sen_len_src mt sen_len_mt** is **mask**	great / error	0.5351	0.9714	0.5263	0.5112
	src tok_len_src mt tok_len_mt is **mask**	great / error	**0.6050**	**0.9751**	**0.5913**	**0.5766**
	src source length: **sen_len_src mt** translation length: **sen_len_mt** is **mask**	great / error	0.5782	0.9728	0.5854	0.5695
	src source length: **tok_len_src mt** translation length: **tok_len_mt** is **mask**	great / error	0.5782	0.9728	0.5854	0.5695
	A **mask** translation of **src tok_len_src** is **mt tok_len_mt**.	good / bad	0.5069	0.9686	0.5203	0.5040
+GMT	**src mt gmt** is **mask**.	great / error	0.5819	0.9692	0.6122	0.5934
	src gmt mt is **mask**.	great / error	**0.6649**	**0.9770**	**0.6815**	**0.6658**
	gmt src mt is **mask**.	great / terrible	0.6064	0.9744	0.6129	0.5972
	src mt? **mask gmt**	yes / no	0.6297	0.9754	0.6406	0.6249
	A **mask** translation of **src** is **mt**. **gmt**	great / terrible	0.5835	0.9710	0.6087	0.5910
		good / bad	0.6275	0.9761	0.6218	0.6070
	mask translation **src mt gmt**	good / bad	0.6426	0.9753	0.6618	0.6454
	mask translation **src gmt mt**	good / bad	0.6457	0.9759	0.6617	0.6457
	A great translation of **src** is **gmt**. **mt** is **mask**.	great / terrible	0.5659	0.9711	0.5846	0.5677

Method	Template	Verbalizer	MCC	F1-NOT	F1-ERR	F1-Multi
+RTT	**src rtt mt** is **mask**.	great / error	0.5891	0.9727	0.6047	0.5882
	src mt? **mask rtt**	yes / no	0.5835	0.9710	0.6087	0.5910
	A **mask** translation of **src** is **mt**. **rtt**	good / bad	0.6139	0.9750	0.6179	0.6024
	rtt A **mask** translation of **src** is **mt**.	good / bad	0.6139	0.9750	0.6179	0.6024
	src mt mask rtt	great / error	**0.6539**	**0.9770**	**0.6614**	**0.6462**
+WT	**src word_src mt** is **mask**	great / error	0.5043	0.9692	0.5085	0.4928
	src mt word_mt is **mask**	great / error	0.4973	0.9686	0.5042	0.4884
	src word_src mt word_mt is **mask**	great / error	0.5604	0.9718	0.5691	0.5530
	A **mask** translation of **src** is **mt**. **word_src word_mt**	great / error	**0.6417**	**0.9765**	**0.6452**	**0.6300**
	word_src word_mt A **mask** translation of **src** is **mt**.	good / bad	0.5268	0.9715	0.5000	0.4857
+Similarity	**src mt sim mask**	great / error	0.5047	0.9662	0.5333	0.5153
	src mt similarity: **sim** is **mask**	great / error	0.6035	0.9744	0.6066	0.5911
		great / terrible	0.5782	0.9728	0.5854	0.5695
	A **mask** translation of **src** is **mt**. The	good / bad	0.4105	0.9650	0.4107	0.3964
	similarity is **sim**.	great / error	**0.6155**	**0.9756**	**0.6034**	**0.5887**
	src mt mask sim	yes / no	0.5659	0.9711	0.5846	0.5677
+GEC	**src mt gec_src** is **mask**	great / error	0.5435	0.9719	0.5310	0.5161
	src mt gec_mt is **mask**	great / error	**0.5523**	**0.9725**	**0.5357**	**0.5210**
	A **mask** translation of **src** is **mt**.	great / terrible	0.5135	0.9691	0.5246	0.5084
	gec_src gec_mt	great / error	0.5417	0.9677	0.5714	0.5530
		good / bad	0.4587	0.9677	0.4505	0.4359
+NER	**src ner_src mt ner_mt** is **mask**	great / error	0.5118	0.9703	0.5000	0.4852
	src mt ner_src ner_mt is **mask**.	great / error	0.4759	0.9682	0.4737	0.4586
	A **mask** translation of **src** is **mt**. **ner_src ner_mt**	good / bad	0.5275	0.9631	0.5641	0.5433
	ner_src ner_mt A **mask** translation of **src** is **mt**.	good / bad	**0.5345**	**0.9707**	**0.5378**	**0.5221**

Table 6: Results of experimenting with various templates and verbalizers according to each method. **src** is the source sentence to input; **mt** is the machine translation sentence to input; **mask** is the mask token; **demo_ok** is the demonstration with a positive label; **demo_bad** is the demonstration with a negative label; **gmt** is the google machine translation; **sim** is the similarity; **gec_**$*$ is the grammar error correction result for $*$ sentence; **rtt** is the round-trip translation; **sen_len_**$*$ is the sentence length of $*$ sentence; **tok_len_**$*$ is the token length of $*$ sentence; **ner** is the named entity recognition; **word_**$*$ is the Results of translation of the word that exists in $*$ sentence; **Verbalizer** is configured as positive/negative.