Towards Precise Localization of Critical Errors in Machine Translation

Dahyun Jung **Sugyeong Eo** **Heuiseok Lim**†

Department of Computer Science, Korea University
{dhaabb55,djtnrud,limhseok}@korea.ac.kr

Abstract

The advent of large language models has experienced a remarkable improvement in the field of machine translation. However, machine translation is still vulnerable to critical meaning deviations, which may incur catastrophic issues in social or ethical contexts. In particular, existing critical error detection primarily focuses on identifying sentence-level errors, leaving the precise localization of such errors within the sentence unaddressed. In this paper, we introduce a new task, word-level critical error detection (WCED), to detect critical errors at a fine-grained level in machine translation sentences. The task aims to identify the parts of a machine translation that contain catastrophic meaning distortions. We hypothesize that the ability to determine errors at the sentence level will positively influence the detection of more granular errors. We propose a sentence-level error detection module to predict which words in a sentence have critical errors. Experimental results demonstrate that our method outperforms existing methodologies and LLM in En-De, Zh-En, En-Ru, and En-Ko. Our method is helpful for determining the fine-grained location of errors. We hope that such studies will improve the capacity to address critical errors adeptly.

1 Introduction

Recent advancements in large language models (LLMs) have significantly improved the performance of machine translation (MT), leading to an increased demand for such systems (Moslem et al., 2023; Zhang et al., 2023; Wang et al., 2023; Xu et al., 2023). Concurrently, the necessity for quality estimation (QE), which automatically evaluates the output of MT systems, has also grown. QE measures the quality of MT outputs based on the source and the MT sentence without a human reference (Specia et al., 2009; Yankovskaya et al., 2019; Fomicheva et al., 2020; Zheng et al., 2021).

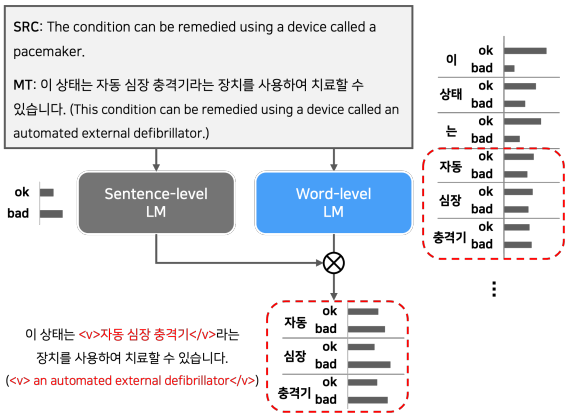

Figure 1: An English to Korean translation example of the WCED and a proposed method. Source (SRC) and MT sentences are given as input to the models. The word-level error detection model predicts a binary label for each representation of an MT word. These probabilities are calculated and combined with the predicted probability of the sentence-level error detection model. The detected errors are presented within <v> and </v> tags.

Despite improvements in MT systems, critical errors remain a challenge (Karpinska and Iyyer, 2023). The QE sub-task, called critical error detection (CED), focuses on identifying instances where critical errors occur (Specia et al., 2021; Zerva et al., 2022). A critical error is a catastrophic meaning distortion with the potential to cause adverse personal or societal impacts (Sudoh et al., 2021; Freitag et al., 2021; Al Sharou and Specia, 2022). Hence, MT outputs must be checked and prevented from containing such critical errors. Although research into CED is conducted in response to this necessity, prior studies primarily concentrate on sentence-level verification to detect the presence of errors in MT sentences (Jiang et al., 2021; Chen et al., 2021; Rubino et al., 2021; Eo et al., 2022). This approach is insufficient as it fails to accurately identify the precise location of errors within sentences, making it challenging to

† Corresponding Author

Findings of the Association for Computational Linguistics: ACL 2024, pages 3000–3012
August 11-16, 2024 ©2024 Association for Computational Linguistics

determine the cause.

To address the issues outlined above, we introduce a new task, word-level critical error detection (WCED), which focuses on identifying segments containing critical errors from the MT sentence. As shown in Figure 1, this task involves analyzing both the source and MT sentences to identify critical errors at the word level within the MT sentence. An example provided illustrates the mistranslation of "pacemaker" as "자동 심장 충격기 (automated external defibrillator)". Such a translation error can lead to significant safety risks for patients due to the dissemination of incorrect information. Through WCED, we are able to identify the specific details of these translation errors, thereby enhancing interpretability.

To predict critical errors at the word level, We propose a method that utilizes a sentence-level detection module. We employ linear interpolation to incorporate the probability representations from the sentence-level detection module into the model calculations for the WCED. This is based on the hypothesis that with a separate module guiding errors within the sentence, errors will be more precisely recognized during the detailed analysis.

In the experiment, we demonstrate that the proposed method evaluates errors with higher accuracy than the baselines. Our model's error detection capability outperforms cutting-edge LLMs such as GPT-3.5 (OpenAI-Blog, 2022) and GPT-4 (OpenAI, 2023). We compare our method with other approaches that utilize the sentence-level detection task, showing that our method is well-suited for the performance enhancement of WCED. Also, we conduct a qualitative analysis and find that our method reacts more sensitively to sentences containing critical errors. By indicating fine-grained critical translation error location, rather than merely detecting the presence of errors in sentences, we provide detailed information about the errors. We expect that incorporating such error detection into MT systems will inform users of potential risk areas. When linked with automatic post-editing, it enables corrections of specific erroneous parts, among other varied applications.

To summarize, our contributions are as follows:

- We enhance the comprehensibility of word-level critical errors by refining the sentence-level CED into a more granular one at the word level.
- We propose a method to improve the performance of the WCED. Our approach aims to increase sensitivity towards errors by utilizing the probability values from the sentence-level detection module.
- The experiments outperform the baseline and demonstrate the feasibility of our method. Therefore, our approach contributes to advancing the task of identifying and correcting critical errors in MT.

2 Related Work

QE was first proposed by Blatz et al. (2004), and its initial studies primarily relied on traditional natural language processing (NLP) techniques (Graham, 2015; Beck et al., 2016). However, the acceleration of deep learning, including the development of neural network architectures like Transformer (Vaswani et al., 2023) and BERT (Devlin et al., 2019), shifts the focus towards developing neural frameworks for QE. DeepQuest (Ive et al., 2018) proposes a framework that adopts a sentence-level approach and generalizes it for document-level QE. OpenKiwi (Kepler et al., 2019) introduces a new open-source QE framework based on bidirectional LSTM. TransQuest (Ranasinghe et al., 2020) leverages a cross-lingual transformer, supporting two different architectures at the sentence level for QE.

The 2021 conference on machine translation (WMT21) introduces the CED as a sub-task for QE (Specia et al., 2021). Critical errors can appear as mistranslations, hallucinations, or deletions. Mistranslation is characterized by the incorrect translation of the source sentence, leading to a distortion of the original meaning. Hallucination refers to the introduction of content not present in the original sentence, while deletion involves the omission of content that is present in the source sentence. Prior research has introduced various methodologies for developing models that perform binary classification of critical errors. These methodologies encompass extracting sentence features (Jiang et al., 2021), employing task-specific classifiers (Chen et al., 2021), altering the architecture of models, and leveraging large volumes of synthetic data (Rubino et al., 2021), as well as incorporating additional information (Eo et al., 2022) to enhance performance.

Previous studies do not perform word-level detection on critical errors. Although existing research concentrates on identifying spans contain-

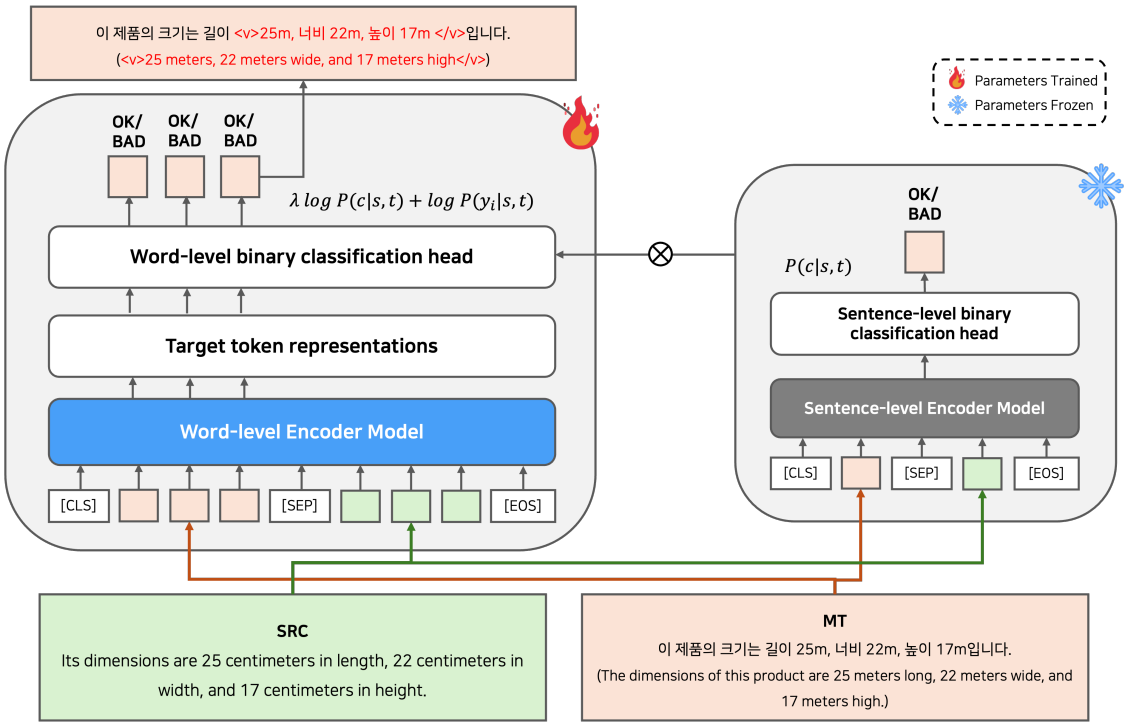

Figure 2: Overview of the proposed method. In this process, the English source sentence and the Korean MT sentence serve as inputs for both word-level and sentence-level error detection models. The word-level model predicts the presence of errors based on the token representations of the MT sentence. The sentence-level model predicts error for the [CLS] token representation. The predictions from the sentence-level model are then combined with those from the word-level, guiding the word-level model.

ing errors (Geng et al., 2023; Guerreiro et al., 2023; Kocmi and Federmann, 2023), these are not specifically designed to detect critical errors. This presents a significant challenge: even if a sentence contains a critical error, it is difficult to pinpoint where the error has occurred. Therefore, we propose the WCED that detects critical errors at the word level. This approach allows for a more detailed and accurate identification of critical errors within MT.

3 Word-level Critical Error Detection (WCED)

In this section, we define the task of word-level critical error detection (WCED) and propose methods to enhance the word-level error detection process. To provide appropriate guidance for exploring errors in a granular manner, we hypothesize an ideal sentence-level CED module capable of accurately identifying the presence of errors at the fine-grained level. Even in the absence of a perfect CED module, leveraging existing modules will be sufficient to assess their effectiveness in approximating our hypothesis.

3.1 Problem Formulation

WCED refers to identifying specific portions of a sentence that contain errors leading to significant changes in meaning, potentially causing confusion or misinterpretation for the reader. This task focuses on detecting critical errors at the word level. To achieve this, the model takes as input a source sentence $s = \{s_1, s_2, ..., s_n\}$ with length n and an MT sentence $t = \{t_1, t_2, ..., t_m\}$ with length m. The model assesses semantic differences between the source and MT sentences to determine words that contain critical errors. Consequently, each word t_i in the MT sentence is associated with a label $y_i \in \{\mathsf{ok}, \mathsf{bad}\}$, where $1 \leq i \leq m$. The label ok indicates that the corresponding word t_i is correctly translated, while bad signifies the presence of critical errors.

3.2 Proposed Method

As depicted in Figure 2, our proposed method involves taking a source sentence s and an MT sentence t as inputs and utilizing a model $P(y_i|s,t)$ to detect errors at the word level. The model concatenates the source and MT sentences using a separator token ([SEP]), and employs an encoder

structure to process this combined input. The encoder outputs a multi-dimensional last hidden state, represented by a d-dimensional vector. Word-level error detection extracts the hidden state corresponding to each token in the MT sentence. It determines whether each word contains an error, thereby providing word-level predictions $\hat{y}_i \in \{\texttt{ok}, \texttt{bad}\}$.

We introduce a sentence-level CED module $P(c|s,t)$ that assesses the presence of critical errors in an MT sentence, adopting a binary classification approach for determining the existence of errors. Specifically, the CED module serves as a guide at the sentence level, providing broad indicators of errors within the sentence. The application of this module enables the WCED model to be more sensitively tuned to errors. This module infers the presence of errors at the sentence level by passing the hidden state of the first token (`[CLS]`) through a binary classification head. The binary variable c is represented as ok when no error is present in the MT sentence and bad when an error exists.

In this structure, leveraging the probability of the CED module $P(c|s,t)$, we derive the final prediction score of error prediction at the word level as follows:

$$\log S(y_i) = \lambda \log P(c|s,t) + \log P(y_i|s,t) \quad (1)$$

where λ represents an additional weight used to adjust the influence of the CED module. By tuning this hyper-parameter, we can finely calibrate the contribution of the CED module to the final error detection score. This formula combines the probability of predicting an error at the word level from the WCED model $P(y|s,t)$ with the probability of the existence of an error in the sentence obtained through the CED module $P(c|s,t)$. The CED module is provided in pre-trained and frozen parameters during this process. The loss function $\mathcal{L}$ is defined as:

$$\mathcal{L} = -\frac{1}{m}\sum_{i=1}^{m} w_{y_i} \log S(y_i) \quad (2)$$

where m denotes the number of tokens in the sentence, and w_{y_i} denotes the class weights given for the ok and bad tags. Through this approach, we present a WCED model that effectively responds to errors and more precisely identifies errors.

3.2.1 Inference

Each word in MT sentences is processed through the token-level linear layers, resulting in a distribution. We calculate the average of these distributions to construct a set of word-level tags, $\hat{y} = \{\hat{y}_1, \hat{y}_2, ..., \hat{y}_m\}$. From these tags, words identified as critical errors are marked with <v> and </v>.

4 Experiments

4.1 Settings

Datasets We leverage datasets from the WMT23 QE (Blain et al., 2023), which are founded on the expert-based multi-dimensional quality metrics (MQM) dataset (Freitag et al., 2021), to collect data for WCED across English-German (En-De), Chinese-English (Zh-En), and English-Russian (En-Ru) language pairs. This dataset defines major errors as those that result in significant alterations to the intended meaning, potentially leading to user misunderstandings. Given the contextual alignment with the definition of critical errors, we selectively extract sentences manifesting major errors from the dataset. We select data comprised exclusively of sentences that have reached a consensus on the presence of such errors and employ it within our experimental framework.

We utilize a dataset for the CED task in the English-Korean (En-Ko) language pair, employing an En-Ko CED dataset[1]. The En-Ko CED dataset comprises target sentences with critical errors and reference sentences. This dataset is designed to intentionally insert critical errors into the reference sentences, allowing for the identification of incorrect words by comparing the target sentences with the reference sentences. After conducting this task with GPT-4 and reviewing the overall results generated, we are confident that the task was executed as intended. We provide statistical details of the datasets in Appendix A.

Implementation Details The implementation is predicated on the COMET GitHub repository[2], with experiments conducted using the large versions of XLM-RoBERTa (Conneau et al., 2020) and InfoXLM (Chi et al., 2021) as the backbone. Training is carried out on an NVIDIA RTX A6000 GPU, with a batch size 32 for 20 epochs. The criterion for early stopping is set at 10 epochs. Performance evaluations of the model are conducted in 2

[1] https://www.aihub.or.kr/
[2] https://github.com/Unbabel/COMET/

epochs during training, with fine-tuning processes taking approximately two hours. The AdamW optimizer is employed for model optimization, setting the learning rate at 1.5e-05 during the fine-tuning. All hyperparameters are manually adjusted. We train a CED module utilizing mBERT (Libovický et al., 2019), XLM-RoBERTa, InfoXLM, and COMETKIWI (Rei et al., 2022). The training settings are consistent with those applied in the training of the WCED model.

Evaluation We adopt F1 score, recall, and precision in alignment with the evaluation criteria set forth by the WMT23 (Blain et al., 2023) error span detection task. Additionally, we incorporate the Matthews correlation coefficient (MCC) (Chicco and Jurman, 2020) as an evaluation metric for sentence-level detection. This metric, previously utilized in the evaluations of WMT21 (Specia et al., 2021) and WMT22 (Zerva et al., 2022) QE tasks, assesses the correlation between actual and predicted error detections.

Baselines Our method is compared against baselines composed of multilingual language models and methods applicable to sentence-level error detection.

- **Fine-Tuning (FT)**: We employ mBERT, XLM-RoBERTa, and InfoXLM, which are multilingual language models, as our baselines, fine-tuning them.

- **Multi-Task Learning (MTL)**: This method aims to enhance the performance of the primary task by simultaneously training on several related tasks (Ruder, 2017). It leverages the shared knowledge gained from various tasks to aid the model in learning more generalized representations. In this paper, we show the performance of conducting CED alongside WCED. The MTL model concurrently predicts sentence-level label c and word-level label y_i. This framework is built upon the COMETKIWI and is capable of executing two tasks. Sentence-level modeling predicts the overall sentence evaluation score using the hidden state of the first input token, whereas word-level modeling assesses the correctness of MT tokens. We convert sentence-level score prediction into sentence-level binary classification for utilization.

- **Soft Prompt Tuning (SPT)**: This approach involves augmenting the model's input with prompts represented in vector representation (Lester et al., 2021; Liu et al., 2022). We incorporate the last hidden state of the CED module into the input embeddings for use as input. This method facilitates knowledge transfer from the CED module to the WCED model, aiding in generating appropriate outputs. We maintain only the added embedding portions as trainable layers while training for the WCED task.

- **Adaptive Prompting (AP)**: The vector representation used in SPT is modified into a form understandable in natural language. This approach involves integrating the results from the CED module into the text input of the WCED model. If the CED module predicts the presence of errors in the source and translated sentences, it adds the word '`terrible`'; otherwise, it adds '`great`' in natural language. This prefix makes the presence of errors in the sentences more explicit, thereby providing a more intuitive reflection of the CED module's outcomes in executing the WCED task.

4.2 Main Results

As shown in Table 1, we report the performance of predicting word-level critical errors. Our proposed method demonstrates enhanced performance for the majority of language pairs when compared to the comparative methods.

In our experiments, we select the InfoXLM and XLM-R models, which demonstrate significantly superior performance among the models considered in FT. Our method is an augmentation to the FT, allowing for direct comparison. Notably, our method achieves remarkable performance with InfoXLM, resulting in F1 scores of 0.4272, 0.4333, 0.7002, and 0.4037 for each language pair, compared to the FT scores of 0.3782, 0.3930, 0.6712, and 0.3492. These improvements can be attributed to utilizing methods within the CED module that provide precise guidelines for error detection.

The application of MTL reveals a performance decline across all language pairs relative to the FT, indicating that conducting sentence-level and word-level error detection simultaneously negatively impacts word-level detection. The results suggest that focusing on a single task within a single model may be more effective.

While the SPT method shows less efficient performance than ours, it still represents an improve-

Method	Model	En-De			Zh-En			En-Ru			En-Ko		
		F1	R	P	F1	R	P	F1	R	P	F1	R	P
Baselines													
FT	mBERT	0.1408	0.1665	0.1475	0.1915	0.2289	0.2058	0.6558	0.6888	0.6503	0.0712	0.0845	0.0734
	InfoXLM	0.4151	**0.4485**	0.4606	0.4052	0.4428	0.4213	0.4355	0.3642	0.6203	0.3898	0.3965	0.4490
	XLM-R	0.3782	0.4106	0.4462	0.3930	0.4332	0.4145	0.6712	0.7206	0.6679	0.3492	0.3457	0.4289
MTL	InfoXLM	0.3536	0.3724	0.4321	0.3079	0.3124	0.3782	0.3866	0.3781	0.5135	0.3684	0.3674	0.4420
	XLM-R	0.3314	0.3069	0.4679	0.2010	0.1738	0.3477	0.4628	0.3871	0.6768	0.2538	0.2839	0.3072
SPT	InfoXLM	0.3797	0.4016	0.4520	0.4208	0.4550	0.4454	0.5882	0.5597	0.6578	0.3616	0.3557	0.4260
	XLM-R	0.4268	0.4418	0.4853	0.3747	0.3813	0.4146	0.6408	0.6898	0.6275	0.3396	0.3538	0.4145
AP	InfoXLM	0.3951	0.4154	0.4798	0.3695	0.3903	0.4194	0.4567	0.4223	0.5831	**0.4303**	0.4332	**0.4915**
	XLM-R	0.3698	0.3684	0.4856	0.3470	0.3641	0.3926	0.6672	0.7116	0.6563	0.3732	0.3960	0.4217
LLMs													
Zero-shot	GPT-3.5	0.0332	0.0276	0.0771	0.0408	0.0388	0.0767	0.0153	0.0108	0.0412	0.0395	0.0319	0.0873
	GPT-4	0.1659	0.2117	0.1867	0.1399	0.1867	0.1276	0.0808	0.0909	0.0758	0.4166	0.4176	0.4600
Few-shot	GPT-3.5	0.0587	0.0687	0.0792	0.0341	0.0525	0.0464	0.0157	0.0138	0.0199	0.1603	0.1663	0.1746
	GPT-4	0.2151	0.2566	0.2292	0.1159	0.1481	0.1204	0.0895	0.0899	0.1047	0.4265	**0.4720**	0.4335
GEMBA	GPT-3.5	0.0827	0.0943	0.1256	0.0413	0.0562	0.0511	0.0226	0.0184	0.0470	0.1491	0.1367	0.1780
	GPT-4	0.2038	0.2199	0.2367	0.1341	0.1614	01392	0.0514	0.0391	0.111	0.3801	0.4062	0.4083
Ours	InfoXLM	**0.4272**	0.4449	**0.4946**	**0.4333**	**0.4652**	0.4463	0.6130	0.6077	0.6634	0.4037	0.4210	0.4660
	XLM-R	0.4008	0.4258	0.4794	0.4232	0.4347	**0.4604**	**0.7002**	**0.7649**	**0.6896**	0.3641	0.4044	0.4364

Table 1: Performance comparison of the proposed method, baseline method, and ChatGPT. We conduct experiments on four language pairs: En-De, Zh-En, En-Ru, and En-Ko. Our proposed method is compared to the following methodologies: fine-tuning (FT), multi-task learning (MTL), soft prompt tuning (SPT), and adaptive prompting (AP). We present the F1 score (F1), Recall (R), and Precision (P) as evaluation metrics. XLM-R is the XLM-RoBERTa model. The best performance is **bold**, and the second is underlined.

ment over the FT. This improvement indicates that the hidden state of the CED module aids word-level detection, underscoring the positive impact of utilizing this module. However, we demonstrate through experimentation that using the representation from the CED module as input is less efficient than our approach, which directly adjusts the error prediction probability.

For the AP, results indicate a decrease in performance compared to FT for the En-De, Zh-En, and En-Ru. This method introduces confusion to the model by providing more definitive natural language outcomes rather than probabilities regarding the presence of errors. Conversely, the performance of the En-Ko language pair is superior with an F1 score of 0.4303 compared to all benchmarked methods. This suggests that providing certainty in sentences where the model is unsure about errors can be beneficial. However, it is important to note that, unlike the proposed method, AP does not consistently show performance improvements across all datasets.

4.3 Comparative Performance with Large Language Models

We experiment to compare the performance of LLMs in the WCED task against our method. We evaluate GPT-4 and GPT-3.5 in both zero-shot and few-shot settings. For GPT-4, we use the 'gpt-4-0125-preview' version; for GPT-3.5, we use the 'gpt-3.5-turbo-0125' version. We include a detailed description of the WCED task in the prompts and provide five demonstrations for the few-shot setting. Appendix B shows the prompts used in this experiment. We further present the performance of GEMBA-MQM (Kocmi and Federmann, 2023), a GPT-based approach for translation quality assessment.

Table 1 shows a comparison between the performance outcomes of ChatGPT and our method. GPT-4 demonstrates superior performance compared to GPT-3.5, showing significant improvements in zero-shot, few-shot scenarios, and GEMBA-MQM. GPT-4 exhibits relatively robust performance in the few-shot setting in the En-De,

	En-De			Zh-En			En-Ru			En-Ko		
	F1	**R**	**P**	**F1**	**R**	**P**	**F1**	**R**	**P**	**F1**	**R**	**P**
ok	0.3213	0.3211	0.4241	0.3958	0.4326	0.4171	0.6084	0.6039	0.6558	0.3517	**0.4068**	0.4278
bad	0.3133	0.3038	0.4281	0.3953	0.4001	0.4508	0.6115	0.6051	0.6588	0.3440	0.3420	0.3936
all	**0.4008**	**0.4258**	**0.4794**	**0.4232**	**0.4347**	**0.4604**	**0.7002**	**0.7649**	**0.6896**	**0.3641**	0.4044	**0.4364**

Table 2: Performance comparison to observe the label-induced probability adjustment effect

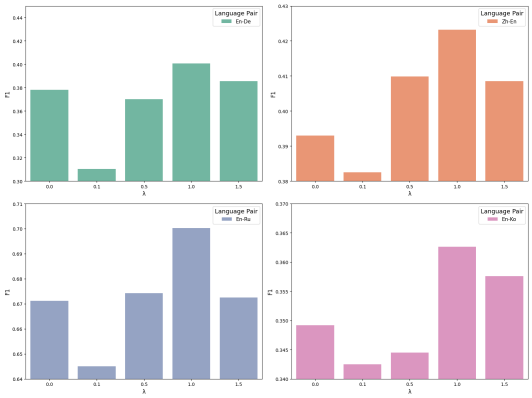

Figure 3: Variation of F1 score with an additional weight λ that adjusts for the impact of the CED module. The model uses XLM-R.

Method	Model	En-De	Zh-En	En-Ru	En-Ko
FT	mBERT	0.4423	0.4349	0.8007	0.1464
	InfoXLM	0.7208	0.6033	0.7601	0.5203
	XLM-R	0.7478	0.6160	0.8007	0.1929
COMET	InfoXLM	0.7137	**0.6318**	0.5178	0.3761
	XLM-R	**0.7481**	0.5796	**0.8204**	**0.6506**

Table 3: MCC performance comparison of CED module for sentence-level error classification

En-Ru, and En-Ko language pairs. However, the examples did not positively impact task comprehension in the Zh-En. Our model significantly outperforms both GPT-3.5 and GPT-4 in the En-De, Zh-En, and En-Ru, indicating that the WCED task presents challenges for LLMs and highlighting the necessity of this work. Even in constrained settings, GPT-4 slightly surpasses the performance of our method in the En-Ko. However, despite its good performance, GPT-4 faces limitations such as cost per token and extended generation times. In contrast, our method is significant in that it can be rapidly utilized for error detection when integrated with MT systems, offering a practical advantage over the computational and financial costs associated with models like GPT-4.

5 Analysis

5.1 The Impact of Weight on CED Module

Figure 3 investigates the impact of λ, a weight that modulates the influence of the CED module, on the model by adding the probability $P(c|s,t)$ from the CED module to $P(y_i|s,t)$ from the WCED model. For all language pairs, performance peaks when λ is set to 1 and diminishes from this point. The improvement in performance when λ is set to 0, where the CED module has no influence, suggests that both excessively low and high weights can negatively affect the model's performance.

5.2 The Performance of CED Module

The performance of the CED module utilized in our experiments is presented in Table 3. Our experiments employ the model with the highest MCC for each language pair. As FT, we use mBERT, InfoXLM, and XLM-R, with the COMET method aiming to evaluate errors at the sentence level through InfoXLM and XLM-R models. Compared to the FT, the COMET demonstrates superior performance in this task. Unlike the word-level error detection, XLM-R generally exhibits higher performance than InfoXLM, particularly for En-Ru and En-Ko, where InfoXLM scores are notably lower at 0.5178 and 0.3761, respectively, compared to the FT scores of 0.7601 and 0.5203. This indicates that while InfoXLM is efficient for word-level detection, XLM-R models are more effective for sentence-level detection.

5.3 The Impact of Using Separate Probabilities for each Label

Table 2 demonstrates the effect of adding the probability from the CED module on the labels. It involves combining the probability values corresponding to the absence of errors (ok) and the presence of errors (bad) from the sum of the probabilities from both the CED module and the WCED model. Thus, the experiment may involve operations that add values to only one of the labels, ok

Input	Output
En-De	
SRC: Thomas Cook's liquidation brought to an end 178 years of solvent trading, prompting the launch of inquiries by the government's Insolvency Service, the accounting watchdog and an influential committee of MPs. **Gold**: Die Liquidation von Thomas Cook beendete 178 Jahre <v>Lösemittelhandel</v> und veranlasste die Einleitung von Untersuchungen durch den Insolvenzdienst der Regierung, den Buchhaltungswächter und ein einflussreiches Komitee von Abgeordneten. (The liquidation of Thomas Cook ended 178 years of <v>solvent trading</v> and led to the initiation of investigations by the government's insolvency service, the accounting officer and an influential committee of deputies.)	**FT**: No critical errors **GPT-4**: 178 Jahre Lösemittelhandel (178 years of solvent trading) **Ours**: Lösemittelhandel und (solvent trading and)
Zh-En	
SRC: 围绕中国的自动驾驶出租车,中国最大网约车服务商滴滴出行发布了年内在上海市郊外启动30辆自动驾驶出租车服务的计划。(China's largest online cab provider DDT has unveiled plans to launch a 30-vehicle self-driving cab service on the outskirts of Shanghai within the year, centering on self-driving cabs in China.) **Gold**: Around China's self-driving taxis, China's <v>largest service provider an</v> nounced a plan to launch 30 self-driving taxi services on the outskirts of Shanghai this year.	**FT**: No critical errors **GPT-4**: an nounced a plan to launch 30 self-driving taxi services on the outskirts of Shanghai this year **Ours**: service provider an nounced
En-Ru	
SRC: And surely, the space around us is ringing after traveling maybe a million light years, or a million years, at the speed of light to get to us. **Gold**: И, конечно же, пространство вокруг нас звенит после того, как <v>мы пролетели, в</v> озможно, миллион световых лет или миллион лет со скоростью света, чтобы добраться до нас. (And, of course, the space around us rings after <v>we've traveled, in</v> perhaps a million light years or a million years at the speed of light to get to us.)	**FT**: в (in) **GPT-4**: звенит послетого, как мы пролетели (rings after we've traveled) **Ours**: мы пролетели, в (we've traveled, in)
En-Ko	
SRC: We have successfully sponsored two polio corrective surgeries so far. **Gold**: 지금까지 두 번의 소아마비 교정 수술을 성공적으로 후원했어요. (We have successfully sponsored two polio corrective surgeries so far.)	**FT**: 소아마비 (polio) **GPT-4**: No critical errors **Ours**: No critical errors

Figure 4: Output examples of the WCED. Gold indicates the correct answer for the span of the MT sentence where the error exists. FT and Ours use InfoXLM, and GPT-4 is the few-shot. If the model finds no errors, we write "No critical errors", and if there are errors, we write only the span where the error exists.

or bad. The conclusion is that applying values to all labels across all language pairs proves superior. It is observed that adjusting values for only one label significantly decreases performance, especially in the En-De and En-Ru pairs. Moreover, for all language pairs except En-Ru, adjustments made to the ok label, indicating the absence of errors, have a more positive impact than adjustments made to the bad label.

5.4 Qualitative Examples

Figure 4 provides examples generated by the model for each language pair. For En-De, the source phrase 'solvent trading' implies a company operating continuously without debt, but its literal translation could be misinterpreted as trading in chemical solvents—an error not detected by the FT. Both GPT-4 and our method identify the erroneous segment, with our method explicitly focusing on the error. While the source mentions a specific company name in the Zh-En, the translation ambiguously renders it. The FT fails to detect this, and GPT-4 extracts an unrelated segment, whereas our model accurately identifies the error's location. For En-Ru, a misinterpretation regarding the subject of travel is noted. Our method accurately predicts this, whereas the FT detects only typographical errors, and GPT-4 includes ranges without errors. The En-Ko example demonstrates that GPT-4 and ours correctly identify sentences without errors, compared to FT. Consequently, our method exhibits a superior ability to detect critical errors compared to GPT-4 accurately and shows greater sensitivity to errors than FT.

6 Conclusion

In this study, we introduced WCED to detect fine-grained critical errors in MT. This task aimed to identify catastrophic meaning distortions at the word level effectively. We proposed a method that utilizes a sentence-level error detection module to predict words containing critical errors within sentences. Experimental results demonstrated that our proposed method significantly enhanced the accuracy of error detection compared to comparison

methods and ChatGPT. These findings indicated that a sentence-level detection module could accurately guide the task of detecting word-level errors. Our approach contributed to a more nuanced evaluation and understanding of MT quality and was crucial in preventing potential issues caused by critical errors. Future research may improve the WCED performance and extend its applicability across various languages and domains. Also, we hope that our methods can be adopted in developing and evaluating MT systems to improve their ability to cope with critical errors.

Limitations

The proposed method exhibits certain limitations in terms of computational efficiency. While employing the CED module, we fix its parameters to reduce computational overhead, leading to increased memory usage compared to conventional models. Also, as the module's output needs to be generated for every sentence during the training process, the total training time extends by the module's inference duration. Nevertheless, once training is complete, it demonstrates advantages in inference costs.

The CED module's applicability is limited as it relies on the same training dataset as the model it is integrated with. To overcome these limitations and achieve similar or improved performance, we can address the issue of generality by enhancing the task with models trained on datasets from different tasks. Future study and development efforts should focus on concretizing and validating these ideas, thereby maximizing the practicality and efficiency of the model. Given the significance of preventing critical errors in MT systems, we hope our work will significantly contribute to advancements in this field.

Ethics Statement

We address the potential negative impact on individuals or society that may arise from erroneous translations, including the possibility of ethical issues related to race, gender, and religion. Our objective is to prevent such severe consequences and enhance the reliability of MT systems.

Acknowledgements

This work was supported by Institute for Information & communications Technology Promotion(IITP) grant funded by the Korea government(MSIT)(RS-2024-00398115, Research on the reliability and coherence of outcomes produced by Generative AI). This research was supported by Basic Science Research Program through the National Research Foundation of Korea(NRF) funded by the Ministry of Education(NRF-2021R1A6A1A03045425). This work was supported by ICT Creative Consilience Program through the Institute of Information & Communications Technology Planning & Evaluation(IITP) grant funded by the Korea government(MSIT)(IITP-2024-2020-0-01819).

References

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.

Daniel Beck, Lucia Specia, and Trevor Cohn. 2016. Exploring prediction uncertainty in machine translation quality estimation.

Frédéric Blain, Chrysoula Zerva, Ricardo Ribeiro, Nuno M Guerreiro, Diptesh Kanojia, José GC de Souza, Beatriz Silva, Tânia Vaz, Yan Jingxuan, Fatemeh Azadi, et al. 2023. Findings of the wmt 2023 shared task on quality estimation. In *Proceedings of the Eighth Conference on Machine Translation*, pages 629–653.

John Blatz, Erin Fitzgerald, George Foster, Simona Gandrabur, Cyril Goutte, Alex Kulesza, Alberto Sanchis, and Nicola Ueffing. 2004. Confidence estimation for machine translation. In *COLING 2004: Proceedings of the 20th International Conference on Computational Linguistics*, pages 315–321, Geneva, Switzerland. COLING.

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.

Zewen Chi, Li Dong, Furu Wei, Nan Yang, Saksham Singhal, Wenhui Wang, Xia Song, Xian-Ling Mao, Heyan Huang, and Ming Zhou. 2021. Infoxlm: An information-theoretic framework for cross-lingual language model pre-training.

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

Alexis Conneau, Kartikay Khandelwal, Naman Goyal, Vishrav Chaudhary, Guillaume Wenzek, Francisco Guzmán, Edouard Grave, Myle Ott, Luke Zettlemoyer, and Veselin Stoyanov. 2020. Unsupervised cross-lingual representation learning at scale.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. BERT: Pre-training of deep bidirectional transformers for language understanding. In *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. 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.

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

Markus Freitag, George Foster, David Grangier, Viresh Ratnakar, Qijun Tan, and Wolfgang Macherey. 2021. Experts, errors, and context: A large-scale study of human evaluation for machine translation. *Transactions of the Association for Computational Linguistics*, 9:1460–1474.

Xiang Geng, Zhejian Lai, Yu Zhang, Shimin Tao, Hao Yang, Jiajun Chen, and Shujian Huang. 2023. Unify word-level and span-level tasks: NJUNLP's participation for the WMT2023 quality estimation shared task. In *Proceedings of the Eighth Conference on Machine Translation*, pages 829–834, Singapore. Association for Computational Linguistics.

Yvette Graham. 2015. Improving evaluation of machine translation quality estimation. In *Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pages 1804–1813.

Nuno M. Guerreiro, Ricardo Rei, Daan van Stigt, Luisa Coheur, Pierre Colombo, and André F. T. Martins. 2023. xcomet: Transparent machine translation evaluation through fine-grained error detection.

Julia Ive, Frédéric Blain, and Lucia Specia. 2018. deepQuest: A framework for neural-based quality estimation. In *Proceedings of the 27th International Conference on Computational Linguistics*, pages 3146–3157, Santa Fe, New Mexico, USA. Association for Computational Linguistics.

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.

Marzena Karpinska and Mohit Iyyer. 2023. Large language models effectively leverage document-level context for literary translation, but critical errors persist. *arXiv preprint arXiv:2304.03245*.

Fabio Kepler, Jonay Trénous, Marcos Treviso, Miguel Vera, and André F. T. Martins. 2019. OpenKiwi: An open source framework for quality estimation. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics: System Demonstrations*, pages 117–122, Florence, Italy. Association for Computational Linguistics.

Tom Kocmi and Christian Federmann. 2023. Gemba-mqm: Detecting translation quality error spans with gpt-4.

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

Jindřich Libovický, Rudolf Rosa, and Alexander Fraser. 2019. How language-neutral is multilingual bert?

Xiao Liu, Kaixuan Ji, Yicheng Fu, Weng Lam Tam, Zhengxiao Du, Zhilin Yang, and Jie Tang. 2022. P-tuning v2: Prompt tuning can be comparable to fine-tuning universally across scales and tasks.

Yasmin Moslem, Rejwanul Haque, John D Kelleher, and Andy Way. 2023. Adaptive machine translation with large language models. *arXiv preprint arXiv:2301.13294*.

OpenAI. 2023. Gpt-4 technical report.

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

Tharindu Ranasinghe, Constantin Orasan, and Ruslan Mitkov. 2020. TransQuest: Translation quality estimation with cross-lingual transformers. In *Proceedings of the 28th International Conference on Computational Linguistics*, pages 5070–5081, Barcelona, Spain (Online). International Committee on Computational Linguistics.

Ricardo Rei, Marcos Treviso, Nuno M Guerreiro, Chrysoula Zerva, Ana C Farinha, Christine Maroti, José GC De Souza, Taisiya Glushkova, Duarte M Alves, Alon Lavie, et al. 2022. Cometkiwi: Ist-unbabel 2022 submission for the quality estimation shared task. *arXiv preprint arXiv:2209.06243*.

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.

Sebastian Ruder. 2017. An overview of multi-task learning in deep neural networks.

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*.

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.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Lukasz Kaiser, and Illia Polosukhin. 2023. Attention is all you need.

Longyue Wang, Chenyang Lyu, Tianbo Ji, Zhirui Zhang, Dian Yu, Shuming Shi, and Zhaopeng Tu. 2023. Document-level machine translation with large language models. *arXiv preprint arXiv:2304.02210*.

Haoran Xu, Young Jin Kim, Amr Sharaf, and Hany Hassan Awadalla. 2023. A paradigm shift in machine translation: Boosting translation performance of large language models. *arXiv preprint arXiv:2309.11674*.

Elizaveta Yankovskaya, Andre Tättar, and Mark Fishel. 2019. Quality estimation and translation metrics via pre-trained word and sentence embeddings. In *Proceedings of the Fourth Conference on Machine Translation (Volume 3: Shared Task Papers, Day 2)*, pages 101–105, Florence, Italy. Association for Computational Linguistics.

Chrysoula Zerva, Frédéric Blain, Ricardo Rei, Piyawat Lertvittayakumjorn, José GC De Souza, Steffen Eger, Diptesh Kanojia, Duarte Alves, Constantin Orăsan, Marina Fomicheva, et al. 2022. Findings of the wmt 2022 shared task on quality estimation. In *Proceedings of the Seventh Conference on Machine Translation (WMT)*, pages 69–99.

Biao Zhang, Barry Haddow, and Alexandra Birch. 2023. Prompting large language model for machine translation: A case study. *arXiv preprint arXiv:2301.07069*.

Yuanhang Zheng, Zhixing Tan, Meng Zhang, Mieradilijiang Maimaiti, Huanbo Luan, Maosong Sun, Qun Liu, and Yang Liu. 2021. Self-supervised quality estimation for machine translation. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 3322–3334, Online and Punta Cana, Dominican Republic. Association for Computational Linguistics.

Language Pairs	Split	Sentences		Tokens	
		all	bad	all	bad
En-De	train	8,041	634	368,011	6,476
	dev	1,000	81	46,932	874
	test	1,000	82	44,257	869
Zh-En	train	8,221	945	337,293	6,832
	dev	1,028	108	42,896	746
	test	1,029	112	43,926	736
En-Ru	train	7,342	330	307,849	2,090
	dev	500	18	20,295	111
	test	500	32	21,477	195
En-Ko	train	7,265	659	186,792	3,892
	dev	500	56	12,786	372
	test	1,000	76	25,138	462

Table 4: Statistical information about the WCED datasets. We present datasets for four language pairs, divided into train, dev, and test set. The table shows the number of sentences in each dataset (all sentences), the number of sentences labeled as having errors (bad sentences), the total number of tokens (all tokens), and the number of tokens containing errors (bad tokens).

A Dataset Details

We propose statistical information about the dataset in Table 4. We follow previous study (Specia et al., 2021) and keep the proportion of labels containing errors at around 10% across all language pairs. The dataset does not include other minor errors.

B Prompt Example for ChatGPT

We provide ChatGPT with the following instructions, which include a definition of a critical error and a detailed description of the task:

```
You are an expert at detecting the span
of critical errors in translations.
Given a source sentence and a
translated sentence, print out the
parts of the entire sentence containing
critical errors, marked with <v> and
</v>. A critical error is a translation
error that completely changes the
meaning of the source text to the
extent that it can have a negative
impact on individuals or society.
Critical errors appear in the following
forms:

- Mistranslation occurs when essential
content is inaccurately translated,
resulting in a change of meaning, or
when it is either not translated at all
(remaining in the source language) or
translated into incomprehensible text.
- Hallucination refers to the addition
of critical content in the translation
that does not exist in the source, such
as the insertion of profanity not
present in the original text.
- Deletion involves the omission of
crucial content from the source
sentence in the translation, which can
include the removal of negations or
offensive words that were present in
the original.

We want to detect words in the
translated sentence with these critical
errors. If there are no critical
errors, print `NONE'. No additional
annotations are required.

Source: {source sentence}
Translation: {target sentence}
```

C Efficiency of CED module in LLM

Table 5 presents the results of experiments conducted by integrating the sentence-level CED module outputs with the LLM baseline. The results demonstrate a notable performance improvement, indicating that the CED module enhanced the performance of the LLM. This aligns with the trends demonstrated in our proposed method, which emphasizes the efficiency of integrating sentence-level features. Nevertheless, the performance indicates that our method, which controls probability during training, is more effective.

		En-De			Zh-En			En-Ru			En-Ko		
Method	**Model**	**F1**	**R**	**P**	**F1**	**R**	**P**	**F1**	**R**	**P**	**F1**	**R**	**P**
w/o CED module													
Zero-shot	GPT-3.5	0.0332	0.0276	0.0771	0.0408	0.0388	0.0767	0.0153	0.0108	0.0412	0.0395	0.0319	0.0873
	GPT-4	0.1659	0.2117	0.1867	0.1399	0.1867	0.1276	0.0808	0.0909	0.0758	0.4166	0.4176	0.4600
Few-shot	GPT-3.5	0.0587	0.0687	0.0792	0.0341	0.0525	0.0464	0.0157	0.0138	0.0199	0.1603	0.1663	0.1746
	GPT-4	0.2151	0.2566	0.2292	0.1159	0.1481	0.1204	0.0895	0.0899	0.1047	0.4265	**0.4720**	0.4335
w/ CED module													
Zero-shot	GPT-3.5	0.1362	0.1206	0.2534	0.0910	0.0677	0.2042	0.1552	0.1130	0.4379	0.1630	0.1446	0.2826
	GPT-4	0.1770	0.2318	0.1778	0.1137	0.1572	0.1019	0.0111	0.0167	0.0083	0.2037	0.2057	0.2382
Few-shot	GPT-3.5	0.1899	0.2411	0.2427	0.1099	0.1250	0.1122	0.0646	0.0571	0.1167	0.2528	0.2893	0.2553
	GPT-4	0.2356	0.2898	0.2575	0.1758	0.2446	0.1622	0.1429	0.1000	0.2500	**0.4390**	0.4640	0.4570
Ours	InfoXLM	**0.4272**	**0.4449**	**0.4946**	**0.4333**	**0.4652**	0.4463	0.6130	0.6077	0.6634	0.4037	0.4210	**0.4660**
	XLM-R	0.4008	0.4258	0.4794	0.4232	0.4347	**0.4604**	**0.7002**	**0.7649**	**0.6896**	0.3641	0.4044	0.4364

Table 5: Comparison of LLM performance with the addition of CED module