mathematics

MDPI

Article

Enhancing Machine Translation Quality Estimation via Fine-Grained Error Analysis and Large Language Model

Dahyun Jung [1], Chanjun Park [2], Sugyeong Eo [1] and Heuiseok Lim [1,*]

[1] Department of Computer Science and Engineering, Korea University, Seoul 02841, Republic of Korea; dhaabb55@korea.ac.kr (D.J.); djtnrud@korea.ac.kr (S.E.)
[2] Upstage, Yongin 16942, Republic of Korea; chanjun.park@upstage.ai
* Correspondence: limhseok@korea.ac.kr

Abstract: Fine-grained error span detection is a sub-task within quality estimation that aims to identify and assess the spans and severity of errors present in translated sentences. In prior quality estimation, the focus has predominantly been on evaluating translations at the sentence and word levels. However, such an approach fails to recognize the severity of specific segments within translated sentences. To the best of our knowledge, this is the first study that concentrates on enhancing models for this fine-grained error span detection task in machine translation. This study introduces a framework that sequentially performs sentence-level error detection, word-level error span extraction, and severity assessment. We present a detailed analysis for each of the methodologies we propose, substantiating the effectiveness of our system, focusing on two language pairs: English-to-German and Chinese-to-English. Our results suggest that task granularity enhances performance and that a prompt-based fine-tuning approach can offer optimal performance in the classification tasks. Furthermore, we demonstrate that employing a large language model to edit the fine-tuned model's output constitutes a top strategy for achieving robust quality estimation performance.

Keywords: natural language processing; quality estimation; fine-grained error span detection

MSC: 68T50

Citation: Jung, D.; Park, C.; Eo, S.; Lim, H. Enhancing Machine Translation Quality Estimation via Fine-Grained Error Analysis and Large Language Model. *Mathematics* **2023**, *11*, 4169. https://doi.org/10.3390/math11194169

Academic Editor: Chengjie Sun

Received: 20 September 2023
Revised: 2 October 2023
Accepted: 3 October 2023
Published: 5 October 2023

1. Introduction

Quality estimation (QE) is the task of evaluating the quality of machine translation (MT) outputs without relying on a gold reference, based solely on the source and the translation output [1,2]. With the growing interest in large language models (LLMs), the significance of QE for measuring MT quality has become increasingly important.

In prior QE studies, the focus has predominantly been on assessing the quality of translations at the sentence or word level. Specifically, even in research that aims to pinpoint quality at a more granular word level, analysis has been limited to determining the presence of errors within words. However, it is essential to elucidate not only the presence of errors in words but also the severity of these errors to provide a more detailed evaluation of the quality in MT sentences.

Fine-grained error span detection is a word-level sub-task first proposed to address this need in WMT 2023 that aims to predict the translation error spans as opposed to binary OK/BAD tasks. This task uses the error spans obtained from the MQM annotations. The task aims to predict the error span (start and end indices) and the error severity (major or minor) for each segment. For example, consider a source sentence, "Don't know where he got the higher price from.", and its corresponding MT sentence, "Er weiß nicht, woher er den höheren Preis bekam. (He doesn't know where he got the higher price.)". In this case, the span can be marked as follows, noting the severity as "minor" for both errors:

- <n>Er weiß nicht</n>, woher er den höheren Preis bekam.

- Er weiß nicht, woher er den höheren Preis <n>bekam</n>.

In this work, we introduce a segmented process that divides the task into three main components: (1) error detection, (2) span extraction, and (3) severity assessment. Error detection is responsible for identifying whether a given sentence contains errors, while the severity assessment categorizes these errors as minor or major. These components are treated as sentence-level binary classification tasks and executed using a prompt-based fine-tuning method. Prompt-based learning effectively leverages pre-trained knowledge by reformulating the task at hand [3–6]. This approach validates QE tasks that determine the presence or absence of critical errors [7]. For the span extraction task in sentences containing errors, we employ word-level QE, which assigns OK/BAD tags to each token in the target translation [8,9]. This method allows us to select the span containing the error.

Subsequently, we utilize an LLM, which has demonstrated remarkable capabilities across various natural language processing (NLP) tasks [10]. These models often cannot be efficiently fine-tuned for specific tasks. Therefore, we propose a paradigm that employs in-context learning to improve the performance of smaller fine-tuned models through post-editing [11].

Our models perform strongly in both English-to-German (En-De) and Chinese-to-English (Zh-En) translations for fine-grained error span detection. These results demonstrate that task granularity can effectively enhance performance. Furthermore, we conduct experiments to augment the outputs of the fine-tuned models using LLM, thereby providing empirical evidence for efficiently utilizing black-box LLM. Our final results demonstrate a significant improvement, with an increase of 0.0755 in the F1 score compared to our baseline. The main contributions of this paper are as follows:

- To our best knowledge, this is the first work to explore the framework of fine-grained span detection.
- We maximize the accuracy of each sub-task by performing task granularity. In addition, we use prompt-based fine-tuning to reduce the gap between pre-training and fine-tuning. Through post-editing, we utilize LLM capabilities to develop the results. All three methods can improve the performance of the model by increasing the accuracy of error detection.
- We conduct extensive experiments on the fine-grained span detection test dataset. The results demonstrate that our framework achieves performance above the baseline.

2. Related Work

Recent remarkable advancements in MT systems have consequently drawn increasing attention to QE for these systems. The field of QE has also experienced rapid growth due to the development of neural network-based architectures such as the Transformer [12] and BERT [13]. While prior research predominantly relied on traditional natural language processing techniques for MT-related studies [14], the acceleration of deep learning has shifted the focus toward developing neural frameworks for QE. DeepQuest proposes a framework that accommodates sentence-level approaches and generalizes them for document-level QE [15]. OpenKiwi introduces a new open-source QE framework based on bidirectional LSTM [8].

XLM-RoBERTa [16] leverages a large-scale multilingual dataset, CommonCrawl [17], to train RoBERTa [18] using masked language modeling (MLM) techniques. This model has demonstrated exceptional performance in cross-lingual tasks, thereby elevating the effectiveness of subsequent QE work. TransQuest offers a straightforward architecture that facilitates training with various types of input (e.g., different language pairs or domains) and enables transfer learning in low-resource settings [19]. COMETKIWI employs a method proposed in IST-Unbabel's WMT 2022 submission paper, integrating the COMET framework with OpenKiwi's predictive estimator structure for sentence- and word-level tasks [20]. Research also exists that proposes self-supervised pre-training using tag-refinement strate-

gies and tree-based annotation techniques to create a human-aligned translation error rate (TER)-based artificial corpus [21].

Despite these paradigmatic shifts, work on quantifying the severity of word-level errors in MT systems remains conspicuously absent. Our research aims to address this gap, offering a more granular approach to quality verification and proposing methodologies to evaluate the quality of MT systems from various perspectives.

3. Methods

In this section, we introduce the applied model architecture for the segmented process (Section 3.1), the prompt-based fine-tuning method for the binary classification tasks (Section 3.3), and strategies for utilizing LLM (Section 3.2). The overall process of our method is presented in Figure 1.

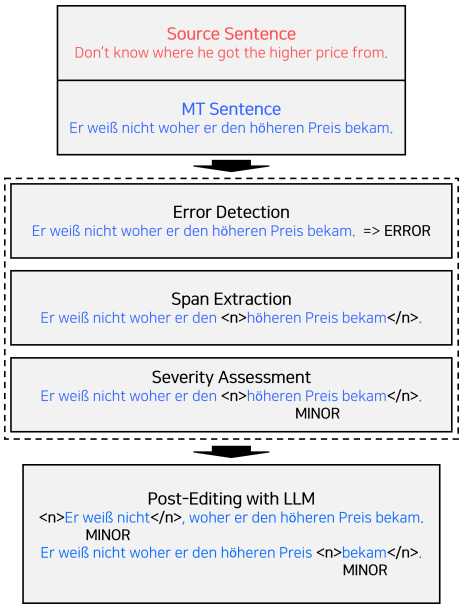

Figure 1. The overall process.

3.1. Task Segmentation

We aim to address the fine-grained error span detection task by segmenting it into distinct sub-tasks and applying suitable model architectures to each.

(1) error detection: To discern the presence or absence of errors at the sentence level, we consider an outcome variable y, which is predicted to be either "error" or "non-error". We utilize both the source sentence and the MT sentence as inputs: x_{error} = <s> $w_1^{src}, \ldots, w_m^{src}$ </s> $w_1^{mt}, \ldots, w_n^{mt}$ </s>, where m and n represent the lengths of the source (src) sentence and the MT (mt) sentence. <s> and </s> are two special tokens to annotate the start and the end of the sentence. The token </s> is also employed as a separator token. To train the model, we utilize the binary cross-entropy loss:

$$\mathcal{L}_{\text{sent}}(\theta) = -(y \log(y_{\text{pred}}) + (1 - y) \log(1 - y_{\text{pred}})) \quad (1)$$

where y is the true label and y_{pred} is the predicted probability of the "error" class.

(2) span extraction: To identify the error span within a sentence containing mistakes, we perform word-level QE. Word-level QE works at a lower granularity level, to predict binary quality labels $y^i \in \{\texttt{OK}, \texttt{BAD}\}$ for all $1 \leq i \leq n$ MT words, indicating whether that word is a translation error. We perform binary classification solely on the tokens of the MT sentence to identify regions predicted as BAD (Figure 2). To train the model for the

word-level QE task, we also use the binary cross-entropy loss function, calculated over all MT words:

$$\mathcal{L}_{\text{word}}(\theta) = -\sum_{i=1}^{n} y^i \log(y^{\text{word}}_{\text{pred},i}) + (1 - y^i) \log(1 - y^{\text{word}}_{\text{pred},i}) \quad (2)$$

where y^i is the true label of the ith MT word, and $y^{\text{word}}_{\text{pred},i}$ is the predicted probability that the ith MT word is `BAD`.

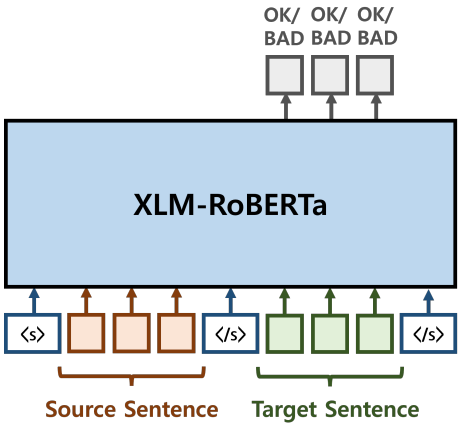

Figure 2. Model architecture of word-level error detection.

(3) severity assessment: We predict the error severity as either "minor" or "major" specifically for sentences where the error span is annotated. This classification is conducted based on the same model architecture used for error presence detection. To formulate the input for the model, we concatenate the source sentence with the MT sentence, which includes the annotated error span: $x_{severity}$ = <s> $w^{src}_1, \ldots, w^{src}_m$ </s> $w^{mt}_1, \ldots,$ <n> $w^{mt}_i, \ldots, w^{mt}_j$ </n>, $\ldots, w^{mt}_{n+2}$ </s>, where i is the start index of error span and j is the end index of error span. The special tokens <n> and </n> demarcate the start and end of the error span.

3.2. Prompt-Based Fine-Tuning

In this task, we design the binary classification 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 [22]. We adopt a pre-trained language model, XLM-RoBERTa, that is trained with MLM objectives. Considering these, we construct a template that reformulates tasks in a cloze style to fill the masked part in the given input text [23,24].

We define x_{prompt} as a form of input that incorporates a template containing `[MASK]` tokens, x_{src} and x_{tgt}. To reduce the overhead associated with prompting in our experiments, we adopt a null prompt for training: x_{prompt} = <s> x_{src} x_{mt} `[MASK]` </s>. The prompt varies based on the template. For the given x_{prompt}, the model is then trained to predict the appropriate word to fill in the `[MASK]` position, such as "great" or "terrible".

Additionally, we introduce a function $v : y \in Y \rightarrow w \in W$ as a function called the verbalizer that maps the label $y \in Y$ to the label word $w_y \in W_Y$. In this case, Y denotes the label set of a targeting task (e.g., $Y = \{\text{NOT}, \text{ERR}\}$ or $\{\text{minor}, \text{major}\}$), and W_Y denotes the corresponding set of label words (e.g., $W_Y = \{\text{great}, \text{terrible}\}$).

3.3. Post-Editing with LLM

We combine the LLM with our fine-tuned smaller model, allowing them to work together to improve performance on the supervised task. We provide the following instructions and two examples to GPT-4 [10] to post-edit the outputs generated by the fine-tuned model:

```
You are an expert in the Fine-grained error span detection task. The
goal of this task is to predict the word-level translation error
spans. You will be asked to predict both the error span (start and
end indices) as well as the error severity (major or minor) for each
segment. There can be multiple error spans, and you must indicate
the severity of the error for the existing spans. If no errors exist
in the translation, the error span is (-1,-1) and the error severity
is no-error.

Review this result by checking the work done by the other workers.
If the work was done correctly, mark it as 'GOOD'; if there were any
errors, re-annotate the Error Span and Error Severity.

To avoid inconsistencies, we expect the indices of the error spans
to correspond to characters in the target string before
tokenization, i.e., the target string that will be provided as
test~data.
```

4. Experiments

4.1. Setting

4.1.1. Datasets

During the fine-tuning, we use the expert-based human evaluation datasets for the submissions of WMT 2020, 2021, and 2022 for En-De and Zh-En [25]. The datasets re-annotated the WMT En-De and Zh-En test sets newstest2020, newstest2021, TED talks, and generalMT2022 with raters that are professional translators and native speakers of the target language (Table 1). In cases where multiple errors exist within a single sentence, the severity of each error is annotated. We analyze the number of errors in each sentence and present the findings in Figure 3.

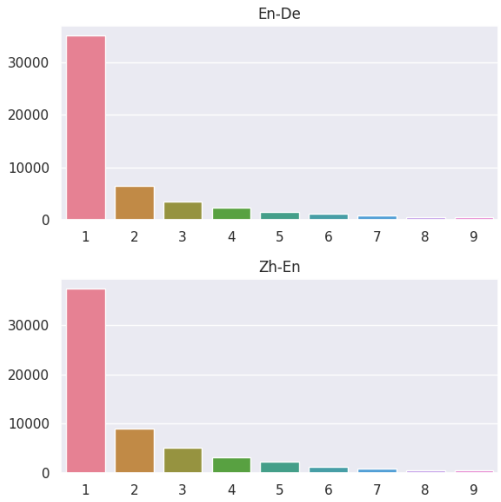

Figure 3. The number of errors contained per sentence, analyzed based on the sentences in which the error exists.

Table 1. Label distribution of the training datasets used by our system.

		Non-Error	Minor	Major
newstest2020	En-De	14,039	55,632	7608
	Zh-En	15,961	57,214	50,074
newstest2021	En-De	5876	3910	2110
	Zh-En	2857	5461	8903
TED talks	En-De	4404	2164	1867
	Zh-En	4297	2266	3352
generalMT2022	En-De	11,240	12,946	3141
	Zh-En	14,415	15,820	15,777

4.1.2. Implementation Details

For training, we choose the large version of XLM-RoBERTa as the backbones of all our models. All models are implemented with PyTorch (https://pytorch.org/, (accessed on 5 September 2023)) and Transformers (https://huggingface.co/, (accessed on 5 September 2023)). We utilize the checkpoints of the pre-trained language model 'xlm-roberta-large'. For sequence-level classification tasks, we use a batch size of 16, the Adam optimizer with a learning rate of 3×10^{-5}, and train for 10 epochs. For the word-level classification task, we use a batch size of 32, the Adam optimizer with a learning rate of 2×10^{-5}, and train for 10 epochs. The experiments are performed on an NVIDIA RTX A6000 environment.

4.1.3. Evaluation Setup

We primarily evaluate our systems regarding the F1 score between the predicted labels and the human annotations for each translation direction (https://wmt-qe-task.github.io/subtasks/task2/, (accessed on 15 September 2023)). Additionally, we present both precision and recall scores for all predictions as part of our system evaluation. We report performance metrics based on predictions made on the official test dataset, for which gold-labeled data are publicly available. The test dataset and evaluation script can be accessed from the WMT 2023 QE Task GitHub repository (https://github.com/WMT-QE-Task/wmt-qe-2023-data, (accessed on 15 September 2023)). The tasks require about 30 min of learning each.

4.2. Results of Detailed Process

The section demonstrates that handling segmented tasks can improve performance. Our proposed model (ED→SE→SA), which tackles tasks sequentially, shows the most promise by outperforming all other models in both F1 score and other metrics (Table 2).

Table 2. Performance comparison for our fine-grained method. We perform an ablation study for each segmented task. ED is Error Detection, SE is Span Extraction, and SA is Severity Assessment. ED+SE+SA performs all detailed tasks simultaneously, while ED→SE→SA performs the tasks sequentially. The highest score is highlighted in bold.

	En-De			Zh-En		
	F1 Score	**Precision**	**Recall**	**F1 Score**	**Precision**	**Recall**
ED+SE+SA	0.1389	0.1723	0.1163	0.1577	0.2592	0.1133
ED→SE+SA	0.1344	0.1711	0.1107	0.1161	0.2638	0.0744
ED+SE→SA	0.1423	0.1435	0.1386	0.1483	0.2619	0.1034
ED→SE→SA	**0.1891**	**0.1989**	**0.1801**	**0.1741**	**0.1896**	**0.1609**

For the En-De task, models performing all three tasks simultaneously (ED+SE+SA) and those executing two tasks concurrently while isolating one (ED→SE+SA and ED+SE→SA) exhibit marginal performance improvements over the baseline. Notably, the model focusing solely on SA within the sequence (ED+SE→SA) demonstrates a significant improvement

in recall, thereby substantiating the importance of the SA model. Our proposed model (ED→SE→SA) outperforms all other models and the baseline. The F1 score jumps to a significantly higher 0.1891, and our model also maintains a balance between precision and recall, indicating robustness.

Our method balances precision and recall better for the Zh-En task, which is evident from the precision score of 0.1896. Although the precision is lower than the baseline, the trade-off produces a better F1 score. Our method achieves the highest F1 score of 0.1741, compared to the baseline score of 0.1555. Furthermore, our method outperforms in terms of both precision and recall, clocking at 0.1896 and 0.1609, respectively. The improved F1 score balances precision and recall, and the positive results from our ablation studies collectively argue in favor of adopting a sequential approach for fine-grained tasks in MT.

4.3. Results of Prompt-Based Fine-Tuning

In this section, we apply and compare two training strategies—conventional fine-tuning and prompt-based fine-tuning—for ED and SA tasks. We evaluate the performance of these strategies using the F1 score as the metric. As indicated in Table 3, the prompt-based fine-tuning approach outperforms conventional fine-tuning in terms of F1 score across both tasks.

Table 3. Results for the error determination and severity assessment tasks. Compare the F1 scores of fine-tuning and prompt-based fine-tuning for those tasks.

		En-De	**Zh-En**
ED	Fine-tuning	0.7473	0.7670
	Prompt-based Fine-tuning	**0.7585**	**0.7740**
SA	Fine-tuning	0.4309	0.6705
	Prompt-based Fine-tuning	**0.4801**	**0.6720**

For the ED task, we achieved an F1 score of 0.7585 for the En-De language pair and 0.774 for the Zh-En pair. We observe similarly high performance in the SA, registering F1 scores of 0.4801 and 0.672, respectively. These findings have several important implications. Firstly, the superior performance of prompt-based fine-tuning suggests its compatibility with our proposed task decomposition strategy. Additionally, enhancing the performance of this approach could further improve the overall efficacy of our system. These results signify that prompt-based fine-tuning can improve task-specific performance without substantially altering the model parameters or data structures. As a result, this approach demonstrates high adaptability and flexibility when applied to new datasets or tasks.

4.4. Results of Post-Editing

In this Table 4, in-context learning (LLM alone) generally demonstrates a lower F1 score for both language pairs. In contrast, our edited fine-tuned models with the LLM exhibit better performance, with F1 scores of 0.2144 and 0.2096 for En-De and Zh-En, respectively.

Table 4. Comparison of the performance of modifying the output of a fine-tuned model using LLM with the performance of LLM alone.

	En-De			**Zh-En**		
	F1 Score	**Precision**	**Recall**	**F1 Score**	**Precision**	**Recall**
In-Context Learning	0.1447	0.2380	0.1040	0.1821	0.1461	0.2418
Edited Fine-tuned Models	**0.2144**	**0.2237**	**0.2058**	**0.2096**	**0.2159**	**0.2037**

The precision and recall figures also back the superiority of the edited fine-tuned models. For instance, in the En-De pairing, the fine-tuned models yield a precision and recall of

0.2237 and 0.2058, notably higher than the 0.238 and 0.104 reported for the in-context learning approach. This suggests not just a general improvement in classification accuracy (as seen in the F1 scores), but also a more balanced performance regarding both false positives and false negatives. Similar trends are observed for the Zh-En language pair. These results indicate that editing fine-tuned models offers a more effective strategy for language learning tasks between these specific language pairs, at least based on the metrics provided.

5. Conclusions

Our approach employed fine-grained error span detection by segmenting tasks and leveraging prompt-based fine-tuning as a robust classification methodology, focusing on two language pairs: En-De and Zh-En. Additionally, we adopted a strategy for LLM-based editing of the output. Through comprehensive experiments and analysis, we demonstrated the efficacy of our system for the given task. Processing tasks sequentially, especially in F1 scores, resulted in a significant performance enhancement by 0.0502 compared to handling tasks concurrently. Additionally, by incorporating prompt-based fine-tuning, we further benefited in the binary classification task. The post-editing approach using LLM improved the F1 score and presented a more balanced precision and recall than the in-context learning method. Our methodology refined quality estimation, allowing for more precise and granular measurements. As part of our future work, we intend to apply our methodology across diverse language pairs and explore its integration with the latest MT strategies.

Author Contributions: Conceptualization, D.J., C.P. and S.E.; methodology, D.J. and C.P.; software, D.J.; validation, D.J.; formal analysis, C.P. and S.E.; investigation, D.J.; resources, H.L.; data curation, D.J.; writing—original draft preparation, D.J.; writing—review and editing, C.P., S.E. and H.L.; visualization, D.J.; supervision, H.L.; project administration, C.P. and H.L.; funding acquisition, H.L. All authors have read and agreed to the published version of the manuscript.

Funding: This work was supported by the 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 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 the 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).

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: All data and the data preprocessing code are available at https://github.com/WMT-QE-Task/wmt-qe-2023-data (accessed on 15 September 2023).

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Specia, L.; Turchi, M.; Cancedda, N.; Cristianini, N.; Dymetman, M. 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, 14–15 May 2009; European Association for Machine Translation: Barcelona, Spain, 2009.
2. Specia, L.; Blain, F.; Fomicheva, M.; Fonseca, E.; Chaudhary, V.; Guzmán, F.; Martins, A.F.T. Findings of the WMT 2020 Shared Task on Quality Estimation. In Proceedings of the Fifth Conference on Machine Translation, Online, 19–20 November 2020; pp. 743–764.
3. Liu, P.; Yuan, W.; Fu, J.; Jiang, Z.; Hayashi, H.; Neubig, G. Pre-train, prompt, and predict: A systematic survey of prompting methods in natural language processing. *ACM Comput. Surv.* **2023**, *55*, 195. [CrossRef]
4. Schick, T.; Schütze, H. 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, EACL 2021, Online, 19–23 April 2021; pp. 255–269.
5. Liu, X.; Zheng, Y.; Du, Z.; Ding, M.; Qian, Y.; Yang, Z.; Tang, J. GPT understands, too. *arXiv* **2021**, arXiv:2103.10385.

6. Brown, T.B.; Mann, B.; Ryder, N.; Subbiah, M.; Kaplan, J.; Dhariwal, P.; Neelakantan, A.; Shyam, P.; Sastry, G.; Askell, A.; et al. Language Models are Few-Shot Learners. *arXiv* **2020**, arXiv:2005.14165.
7. Eo, S.; Park, C.; Moon, H.; Seo, J.; Lim, H. 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), Abu Dhabi, United Arab Emirates, 7–8 December 2022; Association for Computational Linguistics: Abu Dhabi, United Arab Emirates (Hybrid), 2022; pp. 606–614.
8. Kepler, F.; Trénous, J.; Treviso, M.; Vera, M.; Martins, A.F.T. OpenKiwi: An Open Source Framework for Quality Estimation. In Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics: System Demonstrations, Florence, Italy, 28 July–2 August 2019; Association for Computational Linguistics: Florence, Italy, 2019; pp. 117–122.
9. Lee, D. Two-Phase Cross-Lingual Language Model Fine-Tuning for Machine Translation Quality Estimation. In Proceedings of the Fifth Conference on Machine Translation, Online, 19–20 November 2020; pp. 1024–1028.
10. OpenAI. GPT-4 Technical Report. *arXiv* **2023**, arXiv:2303.08774.
11. Xu, C.; Xu, Y.; Wang, S.; Liu, Y.; Zhu, C.; McAuley, J. Small Models are Valuable Plug-ins for Large Language Models. *arXiv* **2023**, arXiv:2305.08848.
12. Vaswani, A.; Shazeer, N.; Parmar, N.; Uszkoreit, J.; Jones, L.; Gomez, A.N.; Kaiser, L.; Polosukhin, I. Attention Is All You Need. *arXiv* **2023**, arXiv:1706.03762.
13. Devlin, J.; Chang, M.W.; Lee, K.; Toutanova, K. BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding. *arXiv* **2019**, arXiv:1810.04805.
14. Specia, L.; Paetzold, G.; Scarton, C. Multi-level Translation Quality Prediction with QuEst++. In Proceedings of the ACL-IJCNLP 2015 System Demonstrations, Beijing, China, 26–31 July 2015; Association for Computational Linguistics and The Asian Federation of Natural Language Processing: Beijing, China, 2015; pp. 115–120.
15. Ive, J.; Blain, F.; Specia, L. deepQuest: A Framework for Neural-based Quality Estimation. In Proceedings of the 27th International Conference on Computational Linguistics, Santa Fe, NM, USA, 20–26 August 2018; Association for Computational Linguistics: Santa Fe, NM, USA, 2018; pp. 3146–3157.
16. Conneau, A.; Khandelwal, K.; Goyal, N.; Chaudhary, V.; Wenzek, G.; Guzmán, F.; Grave, E.; Ott, M.; Zettlemoyer, L.; Stoyanov, V. Unsupervised cross-lingual representation learning at scale. *arXiv* **2019**, arXiv:1911.02116.
17. Wenzek, G.; Lachaux, M.A.; Conneau, A.; Chaudhary, V.; Guzmán, F.; Joulin, A.; Grave, E. CCNet: Extracting High Quality Monolingual Datasets from Web Crawl Data. *arXiv* **2019**, arXiv:1911.00359.
18. Liu, Y.; Ott, M.; Goyal, N.; Du, J.; Joshi, M.; Chen, D.; Levy, O.; Lewis, M.; Zettlemoyer, L.; Stoyanov, V. Roberta: A robustly optimized bert pretraining approach. *arXiv* **2019**, arXiv:1907.11692.
19. Ranasinghe, T.; Orasan, C.; Mitkov, R. TransQuest: Translation Quality Estimation with Cross-lingual Transformers. In Proceedings of the 28th International Conference on Computational Linguistics, Barcelona, Spain, 8–13 December 2020; International Committee on Computational Linguistics: Barcelona, Spain, 2020; pp. 5070–5081.
20. Rei, R.; Treviso, M.; Guerreiro, N.M.; Zerva, C.; Farinha, A.C.; Maroti, C.; de Souza, J.G.C.; Glushkova, T.; Alves, D.M.; Lavie, A.; et al. CometKiwi: IST-Unbabel 2022 Submission for the Quality Estimation Shared Task. *arXiv* **2022**, arXiv:2209.06243.
21. Yang, Z.; Meng, F.; Yan, Y.; Zhou, J. Rethink about the Word-level Quality Estimation for Machine Translation from Human Judgement. *arXiv* **2022**, arXiv:2209.05695.
22. Gao, T.; Fisch, A.; Chen, D. Making pre-trained language models better few-shot learners. *arXiv* **2020**, arXiv:2012.15723.
23. Petroni, F.; Rocktäschel, T.; Riedel, S.; Lewis, P.; Bakhtin, A.; Wu, Y.; Miller, A. 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), Hong Kong, China, 3–7 November 2019; Association for Computational Linguistics: Hong Kong, China, 2019; pp. 2463–2473.
24. Cui, L.; Wu, Y.; Liu, J.; Yang, S.; Zhang, Y. Template-Based Named Entity Recognition Using BART. In Proceedings of the Findings of the Association for Computational Linguistics: ACL-IJCNLP, Online, 1–6 August 2021; pp. 1835–1845.
25. Freitag, M.; Foster, G.; Grangier, D.; Ratnakar, V.; Tan, Q.; Macherey, W. Experts, Errors, and Context: A Large-Scale Study of Human Evaluation for Machine Translation. *arXiv* **2021**, arXiv:2104.14478.