Exploring Effectiveness of GPT-3 in Grammatical Error Correction: A Study on Performance and Controllability in Prompt-Based Methods

Mengsay Loem, Masahiro Kaneko, Sho Takase, and Naoaki Okazaki
Tokyo Institute of Technology
{mengsay.loem, masahiro.kaneko}[at]nlp.c.titech.ac.jp
sho.takase[at]linecorp.com, okazaki[at]c.titech.ac.jp

Abstract

Large-scale pre-trained language models such as GPT-3 have shown remarkable performance across various natural language processing tasks. However, applying prompt-based methods with GPT-3 for Grammatical Error Correction (GEC) tasks and their controllability remains underexplored. Controllability in GEC is crucial for real-world applications, particularly in educational settings, where the ability to tailor feedback according to learner levels and specific error types can significantly enhance the learning process. This paper investigates the performance and controllability of prompt-based methods with GPT-3 for GEC tasks using zero-shot and few-shot setting. We explore the impact of task instructions and examples on GPT-3's output, focusing on controlling aspects such as minimal edits, fluency edits, and learner levels. Our findings demonstrate that GPT-3 could effectively perform GEC tasks, outperforming existing supervised and unsupervised approaches. We also showed that GPT-3 could achieve controllability when appropriate task instructions and examples are given.

1 Introduction

Grammatical Error Correction (GEC) is an essential application of Natural Language Processing (NLP) in educational settings, as it significantly enhances learners' language skills and writing performance (Kaneko et al., 2022). In real-world applications, controlling specific GEC settings, such as minimal and fluency edits and learner level-based corrections, is crucial to address diverse learning needs and scenarios (Napoles et al., 2017; Bryant et al., 2019; Flachs et al., 2020). Although recent GEC approaches based on supervised learning have achieved remarkable progress, they heavily rely on large training datasets comprising both genuine and pseudo data (Xie et al., 2018; Ge et al., 2018; Zhao et al., 2019; Lichtarge et al., 2019; Xu et al., 2019; Choe et al., 2019; Qiu et al., 2019; Grundkiewicz et al., 2019; Kiyono et al., 2019; Grundkiewicz and Junczys-Dowmunt, 2019; Wang and Zheng, 2020; Zhou et al., 2020; Wan et al., 2020; Koyama et al., 2021a). Collecting such data for each specific setting is challenging and time-consuming, which limits the scalability of these methods in various learning situations.

Prompt-based methods utilize large-scale pre-trained language models (PLMs), such as GPT-3, and have demonstrated promising results in numerous NLP downstream tasks. These tasks include natural language inference, question answering, and summarization (Brown et al., 2020; Radford et al., 2019). Given the demand for control in GEC tasks across various settings, prompt-based methods are appealing because they deliver exceptional performance without needing extensive labeled data. Despite the success of prompt-based methods in multiple NLP tasks, their application to GEC remains under-explored. Although Coyne and Sakaguchi (2023) and Fang et al. (2023) have recently assessed prompt-based methods on select GEC benchmarks, a comprehensive analysis has yet to be conducted. This study aims to bridge this gap by concentrating on in-depth analyses of prompt-based methods and their controllability, aspects that have not been thoroughly investigated in previous research.

Our research seeks to address the following questions: 1) To what extent can PLMs using prompt-based methods solve GEC tasks? and 2) Is it possible to control GEC settings with prompts written in natural language using prompt-based methods?

In this work, we demonstrate that prompt-based methods with GPT-3 (Brown et al., 2020) achieve outstanding performance in GEC tasks (Section 3). In addition, the approach provides better control over the GEC process using task instructions and examples (Section 5). We conduct analyses to examine the impact of different types of task instructions on GPT-3's performance in both zero-shot

Proceedings of the 18th Workshop on Innovative Use of NLP for Building Educational Applications (BEA 2023), pages 205–219

and few-shot setting, which emphasizing the importance of appropriate task instructions for GEC tasks (Section 4.1). Additionally, we investigate the effect of varying the number of examples in few-shot setting, and reveal that performance improves as the number of examples increases, albeit not strictly linearly (Section 4.2).

Furthermore, we explore the model's controllability in various GEC scenarios, more specifically, its ability to concentrate on either minimal or fluency aspects (Section 5.1) and edits based on learner levels (Section 5.2). Experimental results indicate that task instructions alone may be sufficient to control editing without examples. However, we found that combining task instructions with examples resulted in more effective controlling performance. This indicates the importance of both task instruction and examples for better control of GEC settings using prompt-based methods, although the example set tends to have more importance.

2 Overall Experimental Settings

In this study, we designed a series of experiments using the prompt-based method with GPT-3 to evaluate the performance in GEC tasks. We utilized the GPT-3 model (`text-davinci-003`) through the API provided by OpenAI[1]. Our experiments were conducted in two settings: zero-shot and few-shot.

Zero-shot In the zero-shot setting, we assessed GPT-3's ability to perform GEC tasks without any prior examples. We employed the following template for prompts in the zero-shot setting:

```
{task instruction}: {input text};
output:___
```

Few-shot For the few-shot setting, we implemented in-context learning as described by Brown et al. (2020). We provided the model with a few examples to guide its understanding of the GEC task. We randomly sampled pairs of examples from the training (or validation) sets of each experimental setting to serve as examples for the model. Details on the number and source of examples used in each experiment are described in the corresponding sections below. The template for prompts in the few-shot setting is as follows:

```
{task instruction}
{example 1}
...
{example N}
{input text}; output:___
```

[1]https://openai.com/blog/openai-api

Prompt We used natural language text prompts for all our experiments. The task instruction within the prompt serves as a directive that informs the model about the desired outcome of each task. We varied the task instructions in both zero-shot and few-shot setting to examine the model's adaptability to different phrasings (refer to Section 4.1). The instruction candidates employed in our prompt analyses are listed in Appendix A. Examples of task instructions include: `Correct the grammatical errors in the following sentence`, `Revise mistakes in this text`, and `Rewrite the following text with proper grammar`.

3 General Performance

To address research question 1) mentioned in Section 1, we investigated the overall performance of the prompt-based method with GPT-3 in GEC tasks. This investigation is particularly relevant given the increasing prevalence of GPT-3 in various NLP applications and the need to assess its potential capabilities for GEC tasks specifically.

3.1 Settings

We evaluated the performance of GPT-3 on three GEC test sets: JFLEG (Napoles et al., 2017), CoNLL2014 (Ng et al., 2014), and W&I+LOCNESS (Bryant et al., 2019; Granger, 1998) using both zero-shot and few-shot settings with 16 examples. We used examples from the training set of JFLEG, NUCLE (Dahlmeier et al., 2013), and W&I+LOCNESS as examples in the few-shot setting when evaluating with JFLEG, CoNLL2014, and W&I+LOCNESS test sets, respectively.

We compared our prompt-based methods to baselines, including supervised and unsupervised approaches. For the supervised approach, we trained a Transformer (big) using the settings described in Vaswani et al. (2017) and employed annotated data from multiple training sets. These sets included W&I+LOCNESS, FCE corpus (Yannakoudakis et al., 2011), Lang-8 Corpus of Learner English (Mizumoto et al., 2012), and NUCLE. After removing uncorrected sentence pairs, the train-

Method	JFLEG	CoNLL2014	W&I+LOCNESS
Transformer (big)	53.22	51.11	51.36
Grundkiewicz and Junczys-Dowmunt (2019)	56.18	44.23	47.89
Grundkiewicz et al. (2019)	–	26.76	–
ChatGPT zero-shot with CoT (Fang et al., 2023)	61.40	51.70	36.10
GPT-3 zero-shot	64.51	56.05	53.07
GPT-3 16-shot	**67.02**	**57.06**	**57.41**

Table 1: Comparison of GPT-3's performance using both supervised and unsupervised approaches on the JFLEG, CoNLL2014, and W&I+LOCNESS test sets in zero-shot and few-shot settings, with 16 examples. The upper block of the table shows the results for the supervised approach, while the middle block shows the results for the unsupervised approaches. The scores are GLEU scores for JFLEG, $F_{0.5}$ scores for CoNLL2014, and W&I+LOCNESS.

ing data used to train the Transformer model was approximately 600K pairs. For unsupervised approach, we compared our methods to previous work in the literature including Grundkiewicz and Junczys-Dowmunt (2019) and Grundkiewicz et al. (2019) where models were pre-trained with synthetic data. We also compared with the result of ChatGPT performance in zero-shot with chain-of-thought (CoT) reported in Fang et al. (2023).

3.2 Results

Table 1 shows the GLEU scores for JFLEG, $F_{0.5}$ scores for CoNLL2014, and W&I+LOCNESS. From the table, GPT-3 performed competitively in the GEC tasks in both zero-shot and few-shot settings, outperforming the Transformer model in all test sets. In the zero-shot setting, GPT-3 surpassed the Transformer, with gains of about 11, 5, and 2 percentage points on JFLEG, CoNLL2014, and W&I+LOCNESS, respectively. The few-shot setting with 16 examples further improved GPT-3's performance, indicating the model's capability to adapt to the task with minimal examples quickly.

When comparing GPT-3 to unsupervised methods, we observe that GPT-3 outperforms other approaches in all test sets consistently. This comparison demonstrates the advantage of GPT-3 over existing unsupervised methods, even in the zero-shot setting. When comparing the performance of ChatGPT in the zero-shot setting with CoT, GPT-3 outperforms ChatGPT CoT in all three test sets. These results indicate GPT-3 is a more effective model for GEC tasks, especially in unsupervised settings.

4 Investigation on Prompt

In this section, we analyze the impact of different factors in prompt on the performance of GPT-3 in GEC tasks. We focus on two factors: (1) the type of task instructions used and (2) the number of examples used in the few-shot settings. Our primary objective is to comprehend the influence of various factors in prompts to the models' output, which will enable us to optimize GPT-3 more effectively for GEC tasks.

4.1 Effect of Task Instruction

In this section, we examine the effect of various types of task instructions on GPT-3's performance in GEC tasks. We conduct evaluations using different task instructions in both zero-shot and few-shot settings.

4.1.1 Settings

We created three types of task instructions, with ten candidates per type, following related work on natural language inference task (Webson and Pavlick, 2022). The types of task instructions are as follows (See Appendix A for details). We used the JFLEG validation set in this experiment.

Instructive instructions explicitly request the model to correct grammatical errors in the given text, such as `Correct grammatical errors in this sentence` and `Revise grammatical mistakes in the following text.`

Misleading instructions do not directly ask for grammar correction but instead require paraphrasing or rewriting, such as `Paraphrase the following sentence` and `Rewrite the following text to make it clearer.`

Irrelevant instructions are unrelated to grammar correction, such as `Translate the following sentence` and `Write a news headline about this sentence.`

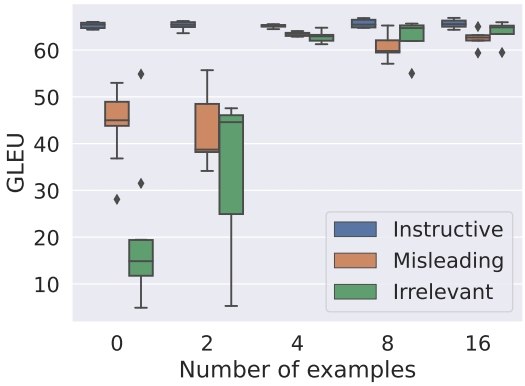

Figure 1: Comparison of GPT-3's performance using different types of task instructions (Instructive, Misleading, and Irrelevant) in zero-shot and few-shot settings on GEC tasks.

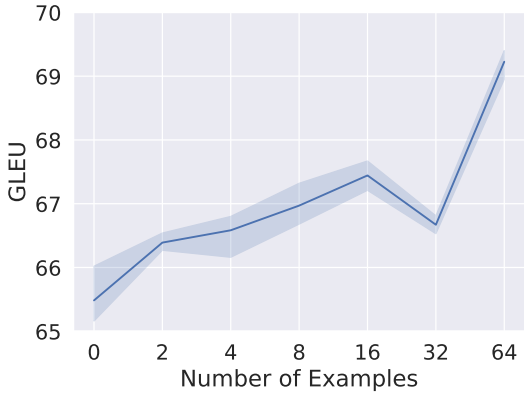

Figure 2: Effect of the number of examples on GPT-3's performance in few-shot settings, evaluated on the JFLEG test set with a fixed task instruction.

4.1.2 Result

Figure 1 shows the summary of the results when using different types of instructions in both zero-shot and few-shot settings. The findings reveal that task instructions significantly affect the performance of GPT-3 in GEC tasks.

In the zero-shot setting, instructive instructions produced the highest average score (65.54), while irrelevant instructions resulted in the lowest average score (17.05), clearly demonstrating that the type of task instruction impacts the model's performance. Misleading instructions fell in the middle, with an average score of 43.45.

In few-shot settings, instructive instructions still outperformed the other two types, but the performance gap between instructive and misleading instructions decreased as the number of examples increased. The variance of the scores decreased with an increasing number of examples, suggesting that the model's performance becomes more consistent as it receives more examples.

When comparing the different few-shot settings, we observed a clear trend of increasing performance as the number of examples increased. The standard deviation also decreased as the number of examples increased, indicating that the model's performance became more consistent with more examples.

4.2 Effect of Number of Examples

In this section, we examine the impact of the number of examples used in few-shot settings on GPT-3's performance. Our objective is to understand how providing varying numbers of examples to the model influences its performance. By maintaining a fixed instruction and focusing solely on varying the number of examples, we aim to better comprehend their effect on the model's performance.

4.2.1 Settings

We conducted experiments on the JFLEG test set to examine the effect of the number of training examples on the model's performance. The task instruction was kept consistent across all experiments. To perform the experiments, we randomly sampled examples from the training set of the JFLEG dataset. We tested the model with 2, 4, 8, 16, 32, and 64 examples, limiting the maximum number of examples to 64 due to the maximum input length of the model employed in our study.

4.2.2 Result

The results obtained from each experimental setting are presented in Figure 2. Our experiments revealed a clear trend: performance improved as the number of examples increased. Our analysis further indicated that the models benefit from having more examples during the few-shot learning process. The highest score of 69.25 was achieved with 64 examples, suggesting that providing more examples can offer better guidance and context for the models to understand and effectively perform the task.

However, it is important to note that performance improvement is not strictly linear with the increase in the number of examples. For instance, the score slightly dipped from 67.11 to 66.67 when the number of examples increased from 16 to 32. This

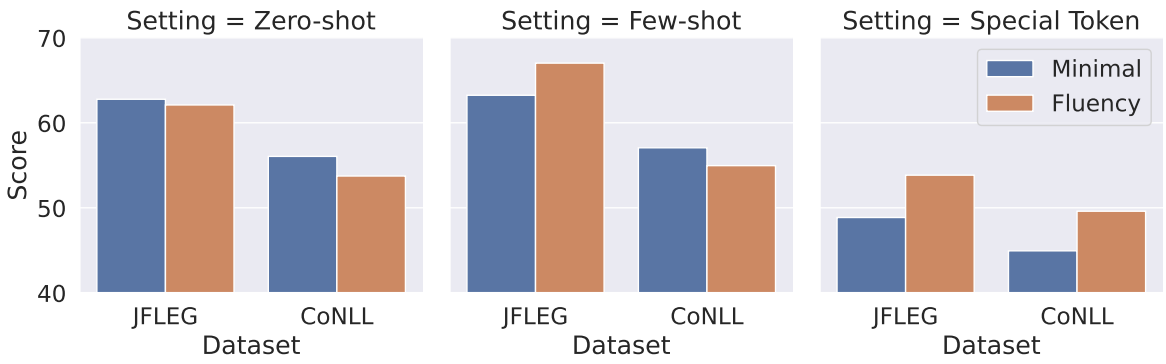

Figure 3: Comparison of GPT-3's controllability for minimal and fluency edits using CoNLL2014 and JFLEG test sets, respectively, measured in GLEU scores.

deviation from linearity could be attributed to the quality of the examples or the inherent variability in the models' performance. Further investigation is required to understand better the factors contributing to these fluctuations and identify the optimal number of examples needed to maximize performance.

5 Controllability through Prompt

In this section, we explore GPT-3's controllability for GEC tasks through prompt-based methods. Our experiments focus on two settings: (1) comparing the model's performance when instructed to make minimal edits versus emphasizing fluency, and (2) tailoring the editing to different learner levels, including beginner, intermediate, advanced, and native speakers. We aim to gain insights into GPT-3's flexibility and controllability under various conditions. We also analyze the relative influence of task instruction and examples to identify the factor that significantly impacts the model's output.

5.1 Minimal vs. Fluency Edits

5.1.1 Settings

We evaluated controllability for minimal and fluency edits using the CoNLL2014 and JFLEG test sets, respectively. CoNLL2014 is a widely-used benchmark for GEC tasks, while JFLEG focuses on fluency-based evaluation. We conducted experiments in zero-shot and 16-shot settings. We used different task instructions to control the settings in the prompts, such as '`Revise the following sentence with proper grammar`' for minimal edits and '`Revise the following sentence to improve fluency`' for fluency edits.

We assessed the models using performance-based evaluation and edit distance-based evaluation. Performance-based evaluation measures the model's error correction or fluency improvement ability, while edit distance-based evaluation quantifies the difference between original and revised sentences, offering insights into the extent of editing performed.

5.1.2 Results

Performance-based Evaluation Figure 3 compares scores in performance-based evaluation for minimal and fluency edit instructions. In the zero-shot setting, minimal edit instructions perform better on the CoNLL2014 test set, while both instructions yield comparable scores on the JFLEG set. In the few-shot setting, higher scores are observed when using corresponding task instructions for each test set, emphasizing the effectiveness of text prompts in controlling editing settings. The discrepancy between zero-shot and few-shot settings might be due to the model's limited understanding of the task in the zero-shot setting. Additional examples in the few-shot setting enable the model to comprehend the task's objective better and adjust its output accordingly.

Additionally, we also compared the prompt-based method with a supervised controlling method that uses special tokens as in Johnson et al. (2017), where different special tokens were used to control target languages in multilingual translation. We trained a Transformer (Big) encoder-decoder with annotated data tagged with special tokens indicating minimal and fluency edits settings. Despite using more training data, this supervised method failed to control specific settings while achieving higher scores on both test sets with fluency edit

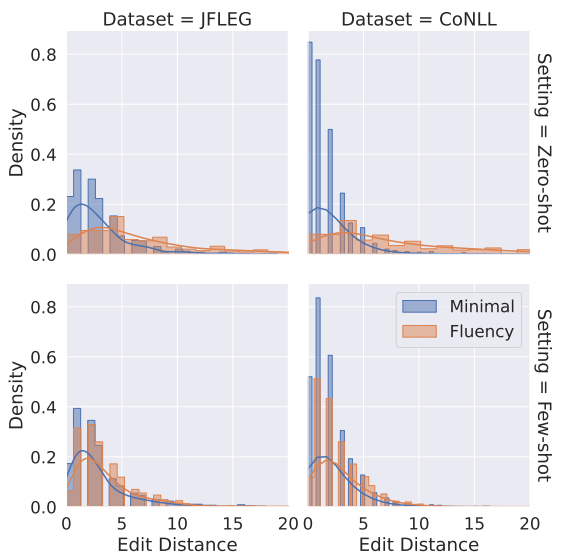

Figure 4: Edit distance distributions for minimal and fluency edits on CoNLL2014 and JFLEG test sets, respectively, as part of the edit distance-based evaluation for controllability of prompts.

tokens, as in Figure 3. This finding highlights the potential advantages of the prompt-based approach.

Edit Distance-based Evaluation Figure 4 presents edit distance distributions for each setting as part of edit distance-based evaluation. A shift to the right indicates more edits performed with fluency edit instructions. In the few-shot setting, the difference in edit distance distributions between minimal and fluency edits is smaller than in the zero-shot setting, which can be attributed to the influence of the examples presented in the prompt. The model's ability to generalize from examples in the few-shot setting may diminish the difference in edit distance between the two settings, further emphasizing the importance of carefully selected examples.

In summary, the prompt-based method using GPT-3 can effectively control GEC task outputs for either minimal or fluency edits. Controllability is more evident in few-shot settings, where additional examples help the model adapt its behavior according to the given instructions. The edit distance-based evaluation further supports the model's ability to adjust its editing behavior based on the prompt, showcasing its potential for practical applications.

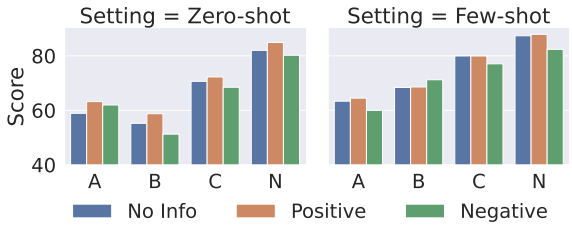

Figure 5: Impact of task instructions with varying additional information on GPT-3's performance in GEC tasks, evaluated on the validation sets of W&I+LOCNESS. The experiment features three settings: No Info, Positive Info, and Negative Info. The x-axis represents different CEFR levels (A, B, C) and native speakers (N) included in the validation set.

5.2 Learner Level-based Correction

5.2.1 Settings

In this section, we examine GPT-3's adaptability to diverse GEC task requirements and contexts by analyzing the impact of varying additional information in task instructions. We conducted experiments in both zero-shot and few-shot (16-shot) settings. We utilized the W&I+LOCNESS validation sets, comprising text from various CEFR levels (A: Beginner, B: Intermediate, C: Advanced) and native speakers (N) as evaluation sets. We devised an experiment with three settings based on the following types of additional information (refer to Appendix B):

No Info: No extra information is provided.

Positive Info: Information that supports the input sentence's characteristics, such as the number of errors to be revised. Example: `"Revise mistakes in the following text written by a beginner learner with a lot of mistakes."`

Negative Info: Information that contrasts with the input sentence's characteristics, e.g., a text written by a beginner learner with many errors but described as having few. Example: `"Revise mistakes in the following text written by an advanced learner with only a few mistakes."`

5.2.2 Results

Figure 5 shows the results of controlling task instruction with additional information on learner levels. In the zero-shot setting, positive information improved performance, while negative information adversely impacted output across most learner levels. This demonstrates the influence of additional

information in task instructions. In the few-shot setting, task instructions without additional information (No Info) achieved comparable scores to cases with Positive Info, suggesting that the model effectively utilizes examples to understand the desired correction level. However, with Negative Info, performance dropped for most learner levels compared to No Info and Positive Info cases.

5.3 Effect of Task Instruction vs. Examples

In this section, we present an experiment to examine the relative effect of task instruction and examples on GPT-3's performance in controllability, in few-shot settings. Our primary objective is to determine which of these two components, task instruction and example, has a more significant impact on the model's outputs. Moreover, we extend our investigation to explore the influence of examples on the editing process of the output, providing a more comprehensive understanding of the interplay between these variables in the context of few-shot learning.

5.3.1 Settings

To investigate the relative influence of task instructions and examples independently, we designed two experiments, each featuring distinct conditions:

Varied Task Instruction with Fixed Examples (VIFE) We modified the task instructions while maintaining a constant set of examples. This approach allows us to assess the influence of task instructions on the model's performance.

Fixed Task Instruction with Varied Examples (FIVE) We utilized a single task instruction and altered the set of examples. This condition helps us evaluate the impact of examples on the model's performance.

In this experiment, we employed the JFLEG and CoNLL2014 test sets. We assessed the performance using $F_{0.5}$ score for CoNLL2014 and GLEU for JFLEG. For the VIFE condition, we prepared a fixed set of examples and a varied set of task instructions for each dataset, similar to the approach in Section 5.1. We used task instructions that requested the model to perform minimal edits on the CoNLL2014 test set and fluency edits on the JFLEG test set. For the FIVE condition, we prepared fixed task instructions and varied examples from the training sets of NUCLE and JFLEG, which correspond to minimal and fluency edits, respectively. We conducted experiments in this section with 16-shot setting.

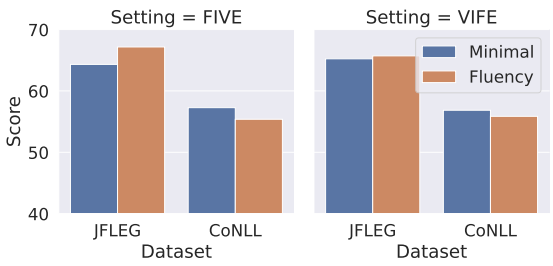

Figure 6: Comparison of the impact of task instructions and number of examples in few-shot settings. VIFE condition examines the effect of varied task instructions with fixed examples, while FIVE condition evaluates the impact of fixed task instructions with varied examples.

Test set	Example from JFLEG	Example from NUCLE
	Fluency Edits	
JFLEG	**0.1569**	0.1893
CoNLL2014	0.4443	**0.4058**
	Minimal Edits	
JFLEG	**0.2283**	0.3038
CoNLL2014	0.4158	**0.3768**

Table 2: Impact of example set on GPT-3's performance in few-shot settings evaluated on JFLEG and CoNLL2014 test sets, measured by Jensen-Shannon distance. Diagonal entries show closer alignment between model output and corresponding example set.

5.3.2 Results

Figure 6 summarizes the results regarding the performance scores. In both CoNLL2014 and JFLEG, we observed performance gaps between the two settings, minimal and fluency edits. However, the gaps were more drastic when changing the example set compared to varying the task instruction. These results suggest that examples play a more critical role in controlling the model's behavior than task instructions, as changing the example set leads to more significant differences in achieving the desired output. This is likely because examples provide specific and contextual information, while task instructions can be abstract and open to interpretation. This highlights the importance of carefully selecting examples to optimize model performance.

We further investigated the example set's impact on model output, using Jensen-Shannon distance to compare edit distance distributions in both minimal

and fluency edits settings. Lower Jensen-Shannon distance indicates a more similar edit distribution between the example set and model output. Results in Table 2 show lower distances in diagonal entries, signifying closer alignment between the model output and corresponding example set. This highlights the importance of carefully selecting examples to guide the model in generating outputs with desired characteristics.

6 Related Work

Supervised learning approaches have predominantly driven GEC research, resulting in state-of-the-art performance. Encoder-decoder models are commonly employed in GEC using supervised learning. Yuan and Briscoe (2016) first applied an encoder-decoder model to GEC, inspiring subsequent researchers to propose various encoder-decoder-based GEC models (Ji et al., 2017; Chollampatt and Ng, 2018; Junczys-Dowmunt et al., 2018; Zhao et al., 2019; Kaneko et al., 2020; Yamashita et al., 2020). These methods typically rely on large training datasets containing parallel sentences with and without grammatical errors (Kiyono et al., 2019). However, scalability remains challenging, as labeled data is required for each specific situation, such as grammar correction style or input text domain.

Unsupervised GEC approaches aim to reduce dependency on labeled data by leveraging unsupervised learning techniques, including PLMs, handcrafted rules, denoising autoencoders, or unsupervised machine translation (Grundkiewicz et al., 2019; Grundkiewicz and Junczys-Dowmunt, 2019; Flachs et al., 2019; Solyman et al., 2021; Koyama et al., 2021b). However, these methods necessitate creating large-scale pseudo data for model training, making it difficult to generate pseudo-data and train models for different learning scenarios. Some studies have proposed unsupervised GEC methods using PLMs (Alikaniotis and Raheja, 2019; Yasunaga et al., 2021), but they have not focused on prompt-based methods with PLMs.

Recently, the GPT-3 model (Brown et al., 2020) has demonstrated remarkable performance across various NLP tasks, although its GEC performance remains limited. Schick et al. (2022) employed a simple zero-shot prompt for GEC, while Dwivedi-Yu et al. (2022) conducted a more comprehensive analysis using diverse zero-shot prompts. Coyne and Sakaguchi (2023) and Fang et al. (2023) compared the latest GPT-3 model's performance (text-davinci-003) and ChatGPT against GEC leaderboard models and reference edits, finding that these prompt-based methods exhibited strong GEC performance. However, automatic metrics and human evaluations occasionally disagreed on the relative quality of corrections.

Controlling GEC model generation is crucial but remains underexplored. Hotate et al. (2019) proposed a GEC method that controls the degree of correction by tagging input with the correction level, but it requires supervised learning with parallel data. Additionally, Hotate et al. (2020) suggested a beam search method to control GEC correction diversity by dynamically updating search tokens within the beam based on the likelihood of predicting source sentence tokens. While this method enables model control without additional training, it falls short in accommodating specific learner requests, such as minimal and fluency edits.

GEC model evaluation methods have been proposed based on learner levels and correction styles. To account for differences in correction styles and domains, Maeda et al. (2022) introduced a method to train evaluation models using only parallel data. Takahashi et al. (2022) created proficiency-annotated data to train evaluation models and developed an evaluation method that considers proficiency by fine-tuning PLMs (Yoshimura et al., 2020).

7 Conclusion

In conclusion, this study demonstrates the potential of using prompt-based methods with GPT-3 for GEC tasks, achieving competitive performance compared to traditional supervised and unsupervised methods. By carefully crafting task instructions and examples, we show that GPT-3 can be effectively controlled to focus on different aspects of the GEC process and adapt to diverse learning needs. Our findings highlight the importance of optimizing task instructions and example selection to enhance the performance and controllability of GPT-3, paving the way for further research on refining prompt engineering techniques and exploring their applicability to other NLP tasks and language models.

8 Educational Implications and Community Benefits

Our study provides valuable implications for education. The controllability of large-scale language models in GEC tasks can be leveraged to design personalized language instruction. It allows educators to provide feedback that matches individual students' proficiency levels and focuses on specific areas for improvement. For learners, instant, tailored feedback can enhance their language learning process. Moreover, our findings can improve intelligent tutoring systems, making them more responsive to individual needs. Beyond education, our research can enhance language-based interfaces and AI communication systems, offering more accurate and context-specific language corrections. This study lays the groundwork for future exploration into how large language models can improve language education and literacy.

9 Limitation

While our study provides valuable insights into the use of prompt-based methods with GPT-3 for GEC tasks and its controllability, several limitations should be acknowledged.

Focus on GPT-3: This study exclusively examines GPT-3 as the language model for GEC tasks. While GPT-3 has shown remarkable performance in various NLP tasks, other pre-trained language models, such as GPT-4, may offer different results. A broader investigation that includes other language models would provide a more comprehensive understanding of the applicability of prompt-based methods in GEC tasks.

Limited evaluation metrics: The evaluation of GPT-3's performance and controllability in our experiments mainly relies on quantitative metrics, such as edit distance and task scores. These metrics may not fully capture the nuances of grammatical error correction or the model's ability to adapt to different learning scenarios. Additional qualitative analysis, along with more diverse evaluation metrics, could provide a richer understanding of the model's performance and controllability.

Variability in examples: While our study highlights the importance of example selection in few-shot settings, we do not thoroughly explore the impact of example quality or diversity. The effect of using different types of examples or a more diverse set of examples remains to be investigated, which could further inform the design of effective example sets for prompt-based GEC tasks. By addressing these limitations in future research, we can further advance our understanding of the performance and controllability of prompt-based methods with GPT-3 and other language models in GEC tasks and beyond.

Potential fine-tuning on test data: There is a possibility that GPT-3 has been fine-tuned (instruction tuning) on the test data we are using, which might explain the higher evaluation scores compared to previous research. As this information has not been disclosed, we are unable to verify it at this time. This point should be taken into consideration when interpreting our results.

Acknowledgements

These research results were obtained partially from the commissioned research (No. 225) by National Institute of Information and Communications Technology (NICT), Japan.

References

Dimitris Alikaniotis and Vipul Raheja. 2019. The unreasonable effectiveness of transformer language models in grammatical error correction. In *Proceedings of the Fourteenth Workshop on Innovative Use of NLP for Building Educational Applications*, pages 127–133, Florence, Italy. Association for Computational Linguistics.

Tom Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared D Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, Sandhini Agarwal, Ariel Herbert-Voss, Gretchen Krueger, Tom Henighan, Rewon Child, Aditya Ramesh, Daniel Ziegler, Jeffrey Wu, Clemens Winter, Chris Hesse, Mark Chen, Eric Sigler, Mateusz Litwin, Scott Gray, Benjamin Chess, Jack Clark, Christopher Berner, Sam McCandlish, Alec Radford, Ilya Sutskever, and Dario Amodei. 2020. Language models are few-shot learners. In *Advances in Neural Information Processing Systems*, volume 33, pages 1877–1901.

Christopher Bryant, Mariano Felice, Øistein E. Andersen, and Ted Briscoe. 2019. The BEA-2019 shared task on grammatical error correction. In *Proceedings of the Fourteenth Workshop on Innovative Use of NLP for Building Educational Applications*, pages 52–75, Florence, Italy. Association for Computational Linguistics.

Yo Joong Choe, Jiyeon Ham, Kyubyong Park, and Yeoil Yoon. 2019. A neural grammatical error correction

system built on better pre-training and sequential transfer learning. In *Proceedings of the Fourteenth Workshop on Innovative Use of NLP for Building Educational Applications*, pages 213–227, Florence, Italy. Association for Computational Linguistics.

Shamil Chollampatt and Hwee Tou Ng. 2018. A Multilayer Convolutional Encoder-Decoder Neural Network for Grammatical Error Correction. In *Proceedings of the Thirty-Second AAAI Conference on Artificial Intelligence*, pages 5755–5762, New Orleans, Louisiana. Association for the Advancement of Artificial Intelligence.

Steven Coyne and Keisuke Sakaguchi. 2023. An analysis of gpt-3's performance in grammatical error correction.

Daniel Dahlmeier, Hwee Tou Ng, and Siew Mei Wu. 2013. Building a large annotated corpus of learner English: The NUS corpus of learner English. In *Proceedings of the Eighth Workshop on Innovative Use of NLP for Building Educational Applications*, pages 22–31, Atlanta, Georgia. Association for Computational Linguistics.

Jane Dwivedi-Yu, Timo Schick, Zhengbao Jiang, Maria Lomeli, Patrick Lewis, Gautier Izacard, Edouard Grave, Sebastian Riedel, and Fabio Petroni. 2022. Editeval: An instruction-based benchmark for text improvements. *ArXiv*, abs/2209.13331.

Tao Fang, Shu Yang, Kaixin Lan, Derek F. Wong, Jinpeng Hu, Lidia S. Chao, and Yue Zhang. 2023. Is chatgpt a highly fluent grammatical error correction system? a comprehensive evaluation.

Simon Flachs, Ophélie Lacroix, and Anders Søgaard. 2019. Noisy channel for low resource grammatical error correction. In *Proceedings of the Fourteenth Workshop on Innovative Use of NLP for Building Educational Applications*, pages 191–196, Florence, Italy. Association for Computational Linguistics.

Simon Flachs, Ophélie Lacroix, Helen Yannakoudakis, Marek Rei, and Anders Søgaard. 2020. Grammatical error correction in low error density domains: A new benchmark and analyses. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 8467–8478, Online. Association for Computational Linguistics.

Tao Ge, Furu Wei, and Ming Zhou. 2018. Fluency boost learning and inference for neural grammatical error correction. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1055–1065, Melbourne, Australia. Association for Computational Linguistics.

Sylviane Granger. 1998. The computer learner corpus: a versatile new source of data for sla research.

Roman Grundkiewicz and Marcin Junczys-Dowmunt. 2019. Minimally-augmented grammatical error correction. In *Proceedings of the 5th Workshop on Noisy User-generated Text (W-NUT 2019)*, pages 357–363, Hong Kong, China. Association for Computational Linguistics.

Roman Grundkiewicz, Marcin Junczys-Dowmunt, and Kenneth Heafield. 2019. Neural grammatical error correction systems with unsupervised pre-training on synthetic data. In *Proceedings of the Fourteenth Workshop on Innovative Use of NLP for Building Educational Applications*, pages 252–263, Florence, Italy. Association for Computational Linguistics.

Kengo Hotate, Masahiro Kaneko, Satoru Katsumata, and Mamoru Komachi. 2019. Controlling grammatical error correction using word edit rate. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics: Student Research Workshop*, pages 149–154, Florence, Italy. Association for Computational Linguistics.

Kengo Hotate, Masahiro Kaneko, and Mamoru Komachi. 2020. Generating diverse corrections with local beam search for grammatical error correction. In *Proceedings of the 28th International Conference on Computational Linguistics*, pages 2132–2137, Barcelona, Spain (Online). International Committee on Computational Linguistics.

Jianshu Ji, Qinlong Wang, Kristina Toutanova, Yongen Gong, Steven Truong, and Jianfeng Gao. 2017. A nested attention neural hybrid model for grammatical error correction. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 753–762, Vancouver, Canada. Association for Computational Linguistics.

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

Marcin Junczys-Dowmunt, Roman Grundkiewicz, Shubha Guha, and Kenneth Heafield. 2018. Approaching neural grammatical error correction as a low-resource machine translation task. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers)*, pages 595–606, New Orleans, Louisiana. Association for Computational Linguistics.

Masahiro Kaneko, Masato Mita, Shun Kiyono, Jun Suzuki, and Kentaro Inui. 2020. Encoder-decoder models can benefit from pre-trained masked language models in grammatical error correction. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 4248–4254, Online. Association for Computational Linguistics.

Masahiro Kaneko, Sho Takase, Ayana Niwa, and Naoaki Okazaki. 2022. Interpretability for language learners

using example-based grammatical error correction. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 7176–7187, Dublin, Ireland. Association for Computational Linguistics.

Shun Kiyono, Jun Suzuki, Masato Mita, Tomoya Mizumoto, and Kentaro Inui. 2019. An empirical study of incorporating pseudo data into grammatical error correction. 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 1236–1242, Hong Kong, China. Association for Computational Linguistics.

Aomi Koyama, Kengo Hotate, Masahiro Kaneko, and Mamoru Komachi. 2021a. Comparison of grammatical error correction using back-translation models. In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Student Research Workshop*, pages 126–135, Online. Association for Computational Linguistics.

Shota Koyama, Hiroya Takamura, and Naoaki Okazaki. 2021b. Various errors improve neural grammatical error correction. In *Proceedings of the 35th Pacific Asia Conference on Language, Information and Computation*, pages 251–261, Shanghai, China. Association for Computational Lingustics.

Jared Lichtarge, Chris Alberti, Shankar Kumar, Noam Shazeer, Niki Parmar, and Simon Tong. 2019. Corpora generation for grammatical error correction. 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 3291–3301, Minneapolis, Minnesota. Association for Computational Linguistics.

Koki Maeda, Masahiro Kaneko, and Naoaki Okazaki. 2022. IMPARA: Impact-based metric for GEC using parallel data. In *Proceedings of the 29th International Conference on Computational Linguistics*, pages 3578–3588, Gyeongju, Republic of Korea. International Committee on Computational Linguistics.

Tomoya Mizumoto, Yuta Hayashibe, Mamoru Komachi, Masaaki Nagata, and Yuji Matsumoto. 2012. The effect of learner corpus size in grammatical error correction of ESL writings. In *Proceedings of COLING 2012: Posters*, pages 863–872, Mumbai, India. The COLING 2012 Organizing Committee.

Courtney Napoles, Keisuke Sakaguchi, and Joel Tetreault. 2017. JFLEG: A fluency corpus and benchmark for grammatical error correction. In *Proceedings of the 15th Conference of the European Chapter of the Association for Computational Linguistics: Volume 2, Short Papers*, pages 229–234, Valencia, Spain. Association for Computational Linguistics.

Hwee Tou Ng, Siew Mei Wu, Ted Briscoe, Christian Hadiwinoto, Raymond Hendy Susanto, and Christopher Bryant. 2014. The CoNLL-2014 shared task on grammatical error correction. In *Proceedings of the Eighteenth Conference on Computational Natural Language Learning: Shared Task*, pages 1–14, Baltimore, Maryland. Association for Computational Linguistics.

Mengyang Qiu, Xuejiao Chen, Maggie Liu, Krishna Parvathala, Apurva Patil, and Jungyeul Park. 2019. Improving precision of grammatical error correction with a cheat sheet. In *Proceedings of the Fourteenth Workshop on Innovative Use of NLP for Building Educational Applications*, pages 240–245, Florence, Italy. Association for Computational Linguistics.

Alec Radford, Jeff Wu, Rewon Child, David Luan, Dario Amodei, and Ilya Sutskever. 2019. Language models are unsupervised multitask learners.

Timo Schick, Jane Dwivedi-Yu, Zhengbao Jiang, Fabio Petroni, Patrick Lewis, Gautier Izacard, Qingfei You, Christoforos Nalmpantis, Edouard Grave, and Sebastian Riedel. 2022. Peer: A collaborative language model.

Aiman Solyman, Wang Zhenyu, Tao Qian, Arafat Abdulgader Mohammed Elhag, Muhammad Toseef, and Zeinab Aleibeid. 2021. Synthetic data with neural machine translation for automatic correction in arabic grammar. *Egyptian Informatics Journal*, 22(3):303–315.

Yujin Takahashi, Masahiro Kaneko, Masato Mita, and Mamoru Komachi. 2022. ProQE: Proficiency-wise quality estimation dataset for grammatical error correction. In *Proceedings of the Thirteenth Language Resources and Evaluation Conference*, pages 5994–6000, Marseille, France. European Language Resources Association.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Ł ukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. In *Advances in Neural Information Processing Systems*, volume 30. Curran Associates, Inc.

Zhaohong Wan, Xiaojun Wan, and Wenguang Wang. 2020. Improving grammatical error correction with data augmentation by editing latent representation. In *Proceedings of the 28th International Conference on Computational Linguistics*, pages 2202–2212, Barcelona, Spain (Online). International Committee on Computational Linguistics.

Lihao Wang and Xiaoqing Zheng. 2020. Improving grammatical error correction models with purpose-built adversarial examples. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 2858–2869, Online. Association for Computational Linguistics.

Albert Webson and Ellie Pavlick. 2022. Do prompt-based models really understand the meaning of their

prompts? In *Proceedings of the 2022 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 2300–2344, Seattle, United States. Association for Computational Linguistics.

Ziang Xie, Guillaume Genthial, Stanley Xie, Andrew Ng, and Dan Jurafsky. 2018. Noising and denoising natural language: Diverse backtranslation for grammar correction. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers)*, pages 619–628, New Orleans, Louisiana. Association for Computational Linguistics.

Shuyao Xu, Jiehao Zhang, Jin Chen, and Long Qin. 2019. Erroneous data generation for grammatical error correction. In *Proceedings of the Fourteenth Workshop on Innovative Use of NLP for Building Educational Applications*, pages 149–158, Florence, Italy. Association for Computational Linguistics.

Ikumi Yamashita, Satoru Katsumata, Masahiro Kaneko, Aizhan Imankulova, and Mamoru Komachi. 2020. Cross-lingual transfer learning for grammatical error correction. In *Proceedings of the 28th International Conference on Computational Linguistics*, pages 4704–4715, Barcelona, Spain (Online). International Committee on Computational Linguistics.

Helen Yannakoudakis, Ted Briscoe, and Ben Medlock. 2011. A new dataset and method for automatically grading ESOL texts. In *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies*, pages 180–189, Portland, Oregon, USA. Association for Computational Linguistics.

Michihiro Yasunaga, Jure Leskovec, and Percy Liang. 2021. LM-critic: Language models for unsupervised grammatical error correction. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 7752–7763, Online and Punta Cana, Dominican Republic. Association for Computational Linguistics.

Ryoma Yoshimura, Masahiro Kaneko, Tomoyuki Kajiwara, and Mamoru Komachi. 2020. SOME: Reference-less sub-metrics optimized for manual evaluations of grammatical error correction. In *Proceedings of the 28th International Conference on Computational Linguistics*, pages 6516–6522, Barcelona, Spain (Online). International Committee on Computational Linguistics.

Zheng Yuan and Ted Briscoe. 2016. Grammatical error correction using neural machine translation. In *Proceedings of the 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 380–386, San Diego, California. Association for Computational Linguistics.

Wei Zhao, Liang Wang, Kewei Shen, Ruoyu Jia, and Jingming Liu. 2019. Improving grammatical error correction via pre-training a copy-augmented architecture with unlabeled data. 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 156–165, Minneapolis, Minnesota. Association for Computational Linguistics.

Wangchunshu Zhou, Tao Ge, Chang Mu, Ke Xu, Furu Wei, and Ming Zhou. 2020. Improving grammatical error correction with machine translation pairs. In *Findings of the Association for Computational Linguistics: EMNLP 2020*, pages 318–328, Online. Association for Computational Linguistics.

A Prompts for Investigation on Instruction Effect

All instructions used for experiments described in Section 4.1 are listed in Table 3.

B Prompts for Learner's Level-based Control

All instructions and additional information used for experiments described in Section 5.2 are listed in Table 4.

Type	Task Instruction
Instructive	Correct grammatical errors in this sentence Revise grammatical mistakes in the following text. Edit this paragraph for grammar mistakes. Find and fix any errors in this sentence. Rewrite this sentence to correct its grammar. Identify and correct the grammar errors in this text. Make any necessary grammar corrections to this passage. Correct the grammar in this sentence without changing its meaning. Find and correct the errors in this paragraph. Proofread this text and correct any grammar mistakes.
Misleading	Paraphrase the following sentence. Rewrite the following text to make it clearer. Revise this paragraph to improve its clarity. Clarify the meaning of this sentence by rephrasing it. Make this sentence more concise without changing its meaning. Improve the readability of this text by rewording it. Reconstruct this sentence to enhance its clarity. Paraphrase this text to make it more comprehensible. Rewrite this paragraph to convey the same information in a clearer way. Edit this sentence to improve its coherence and flow.
Irrelevant	Translate the following sentence in to Japanese. Write a news headline about this sentence. Create a meme based on the following text. Write a short story based on this sentence. Compose a poem using the words in this paragraph. Write a summary of this text. Analyze the use of metaphor in this sentence. Explain the historical context of this passage. Write a tweet about this text. Write a letter to your future self based on the following sentence.

Table 3: Prompts for Instruction Effect Investigation, showing three types of task instructions with ten candidate prompts each. The types include Instructive, Misleading, and Irrelevant prompts.

Info	Task Instruction
Beginner	
No Info	Revise mistakes in the following text
Positive Info	Revise mistakes in the following text written by a beginner learner with a lot of mistakes
Negative Info	Revise mistakes in the following text written by an advanced learner with only a few mistakes
Intermediate	
No Info	Revise mistakes in the following text
Positive Info	Revise mistakes in the following text written by an intermediate learner with some mistakes
Negative Info	Revise mistakes in the following text written by a native speaker
Advanced	
No Info	Revise mistakes in the following text
Positive Info	Revise mistakes in the following text written by an advanced learner with only a few mistakes
Negative Info	Revise mistakes in the following text written by a beginner learner with a lot of mistakes
Native	
No Info	Revise mistakes in the following text
Positive Info	Revise mistakes in the following text written by a native speaker
Negative Info	Revise mistakes in the following text written by a beginner learner with a lot of mistakes

Table 4: All prompts used in experiments investigating the controllability of learner level-based edits.