Deliberate then Generate: Enhanced Prompting Framework for Text Generation

Anonymous ACL submission

Abstract

Large language models (LLMs) have shown remarkable success across a wide range of natural language generation tasks, where proper prompt designs make great impacts. While existing prompting methods are normally restricted to providing correct information, in this paper, we encourage the model to deliberate by proposing a novel Deliberate then Generate (DTG) prompting framework, which consists of error detection instructions and candidates that may contain errors. DTG is a simple yet effective technique that can be applied to various text generation tasks with minimal modifications. We conduct extensive experiments on 20+ datasets across 7 text generation tasks, including summarization, translation, dialogue, and more. We show that DTG consistently outperforms existing prompting methods and achieves state-of-the-art performance on multiple text generation tasks. We also provide in-depth analyses to reveal the underlying mechanisms of DTG, which may inspire future research on prompting for LLMs.

1 Introduction

Large language models (LLMs) (Brown et al., 2020; OpenAI, 2023; Touvron et al., 2023) are revolutionizing the area of natural language generation, which have demonstrated exceptional abilities in generating coherent and fluent text as well as exhibited a remarkable aptitude in performing a diverse range of text generation tasks with high accuracy (Hendy et al., 2023; Nori et al., 2023). When adapting to downstream tasks, traditional fine-tuning methods require access to the parameters of LLMs, which hinder their application on powerful black-box LLMs (e.g., ChatGPT) that only provide APIs to interact with. Therefore, prompting methods that guide the generation results by providing several task-specific instructions and demonstrations have attracted lots of attention in recent works (Schick and Schütze, 2020; Sanh et al., 2021), which show that the prompt can significantly influence the resulting outcomes and thus require careful design.

While prompting is itself a general approach, the current use of this approach is a bit rigid, say, an LLM only operates on the basis of what is correct (Brown et al., 2020; Hendy et al., 2023; Wei et al., 2022b). This is not the case for language acquisition where a human can learn from both positive and negative feedback and improve the ability of language use through corrections. In this work, we examine whether and how the deliberation ability emerges by asking the LLMs to rethink and learn to detect potential errors in their output. To do this, we develop a new prompting template termed Deliberate then Generate (DTG) that contains instructions and candidate outputs to enable an error detection process before generation, i.e., adding "*Please detect the error type firstly, and provide the refined results then*" in the prompt.

A key design aspect of DTG is how to determine the candidate. One straightforward choice is utilizing the results from an extra baseline system, which typically exhibits high quality and requires only minor adjustments. Accordingly, it cannot well facilitate the deliberation ability. In this work, we propose to utilize the text that is irrelevant from the reference (e.g., such as a randomly sampled text or even an *empty string*) as the candidate. In this way, the method successfully triggers the deliberation ability of LLMs, without having to resort to other text generation systems to create correction examples, which enables DTG to be easily applied to a wide range of text generation tasks only with minimal modifications in prompts. This work is in part motivated from a psychological perspective by considering *negative evidence* in developing language abilities, which is a canonical case for language learning (Marcus, 1993).

We conduct extensive experiments on 7 text generation tasks and more than 20 datasets on

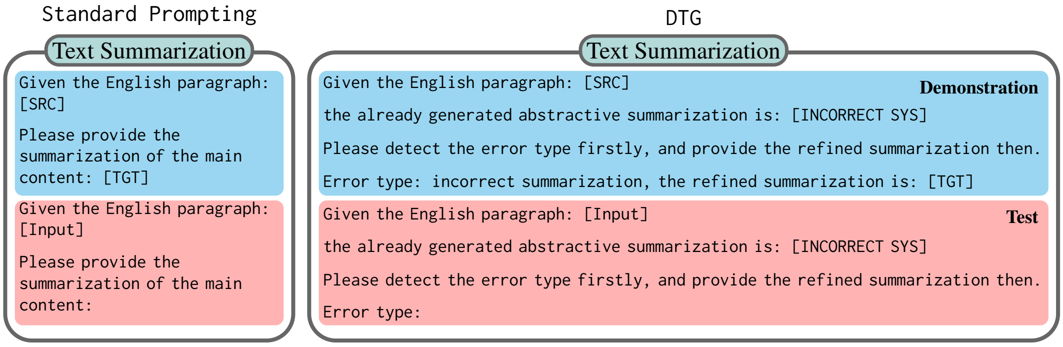

Figure 1: Comparison of standard GPT prompting and our DTG prompt desgin for summarization task. Note that prompt in blue denotes the demonstration, and that in red denotes the test input. [SRC] and [Input] means the source input, TGT means the target reference and [INCORRECT SYS] means the irrelevant system output (e.g., such as a randomly sampled text or even an empty string).

GPT3.5 (`text-davinci-003`) and GPT4, where the proposed DTG prompting consistently improves model performance compared to conventional prompts. GPT with DTG prompting achieves state-of-the-art performance on multiple datasets across different text generation tasks, including machine translation, simplification and commonsense generation. Extensive ablation studies and error statistical analysis illustrate that the proposed DTG prompting does enable deliberation ability and error avoidance before generation.

The main contributions of this work are summarized as follows:

- We propose a novel prompting framework named DTG for LLMs, which eliminates the need for extra resources or costs and can be effortlessly applied to various text generation tasks. DTG can also be combined with other advanced prompting strategy (e.g., CoT) to further improve the performance.
- We conduct experiments on 20+ datasets across 7 text generation tasks, where DTG prompting brings consistent improvements and achieves SoTA performance on several benchmarks.
- To the best of our knowledge, we are the first to evaluate the performance of GPT3.5 and GPT4 on multiple benchmark text generation tasks. We hope the experimental results help deepen our understanding of SoTA LLMs.

2 Related Work

Large Language Models. With the scaling of model and corpus sizes, Large Language Models (LLMs) (Devlin et al., 2018; Radford et al., 2019; Lewis et al., 2019) have achieved remarkable success in various areas of natural language processing. To tailor a model for particular tasks, one approach is to fine-tune it with task-specific datasets (Jiao et al., 2023; Li and Liang, 2021; Hu et al., 2021). Jiao et al. (2023) introduce data with error annotations in fine-tuning to improve the machine translation abilities of open-source LLMs. The fine-tuning approach poses a challenge when applied to powerful black-box LLMs that only offer APIs for interaction, as it requires access to the underlying parameters. With the help of instruction tuning (Wei et al., 2021) and reinforcement learning from human feedback (Ouyang et al., 2022), recent LLMs can achieve gradient-free adaptation to various downstream tasks by prompting with natural language instructions, and some powerful capacities such as in-context learning (Brown et al., 2020) have also emerged.

Prompting Methods. Prompting is a general method for humans to interact with LLMs, which is usually designed as an instruction for a task that guides LLMs toward intended outputs (Schick and Schütze, 2020; Sanh et al., 2021). To make the most of LLMs on downstream tasks, the prompts need to be carefully designed, either manually (Hendy et al., 2023) or automatically (Gao et al., 2020; Zhou et al., 2022). Prompting also provides a way to interact with LLMs in natural language, such as letting them utilize external tools (Schick et al., 2023), resources (Ghazvininejad et al., 2023) and models (Wu et al., 2023; Shen et al., 2023), or conducting Chain-of-Thought (CoT) reasoning in generation (Wei et al., 2022a; Kojima et al., 2022). A concurrent work incorporates answers in pre-

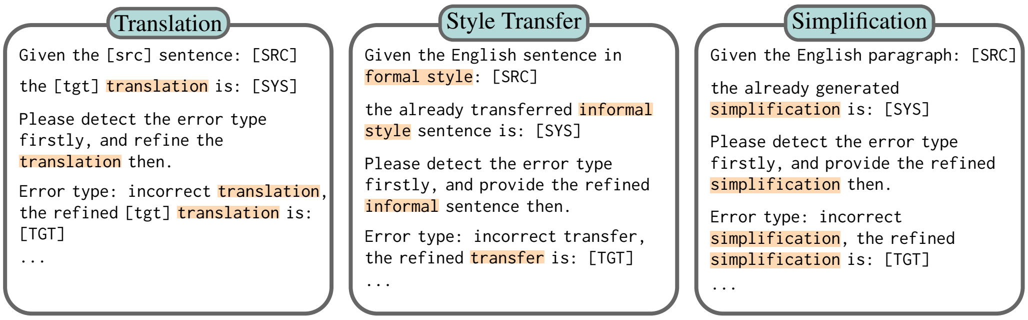

Figure 2: Illustration of DTG demonstration design for machine translation, style transfer and text simplification tasks. Due to the limited page, please refer to the Appendix for the remained 3 generation tasks, including dialogue summarization, paraphrase and commonsense generation.

vious rounds into prompts in an iterative process to improve the accuracy of LLMs on reasoning tasks (Zheng et al., 2023). Besides multi-step reasoning, basic prompts are still widely utilized in general text generation tasks such as machine translation and summarization, where previous advanced methods such as CoT have been shown ineffective (Peng et al., 2023). In this paper, we propose a simple and general prompting method that consistently improves model performance across various text generation tasks, without any additional resources or costs.

3 Deliberate then Generate

Language acquisition by a human is normally based on both positive and negative feedback and improves the ability of language use through corrections. Inspired by this, unlike the conventional prompts only with correct information, we introduce a more deliberate approach termed Deliberate then Generate (DTG) prompting by facilitating LLMs to *detect errors on a synthesized text that may contain errors*. Specifically, the proposed DTG method unfolds in the following manner: 1) It begins with a concise and explicit instruction of the desired task, providing guidance on generating an intended text based on a given input text; 2) A synthesized text is then provided as a candidate output; 3) Finally, DTG encourages the model to detect potential errors, and subsequently generate an improved output after thorough deliberation.

Figure 1 illustrates a comparison between standard prompting and our proposed DTG prompting for the summarization task in the one-shot scenario. A distinctive feature of DTG is its emphasis on error detection other than immediate response. Instead of generating the outcome directly from the

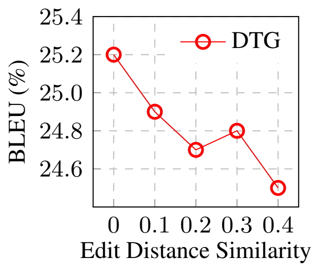

Figure 3: BLEU scores against the similarity (Edit Distance) on ZH-EN task.

given input text, DTG steers the model to make deliberate decisions by detecting the error type firstly based on both the input text, denoted as “[SRC]”, and a pre-defined candidate, denoted as “[SYS]”, before the final decisions. This deliberative process forms the bedrock of the DTG approach and will be further elaborated upon in the analysis section (i.e., Section 6). Besides, a few demonstrations can be provided, imbuing LLMs with an awareness of the expected output (highlighted in blue), and the test input (marked in red). DTG is a general prompting method that could be easily applied to any text generation task with minimal modifications to the prompt. Figure 2 illustrates the particular prompts used for 3 generation tasks we considered, indicating that minimal customization is required across different tasks as highlighted in yellow.

The determination of the synthesized text is another key part of DTG. Straightforwardly, using the output of a baseline system, which can either be LLMs themselves or any other models, is a natural choice. However, such baseline text just requires minor modifications, and thus cannot well trigger the deliberation ability of LLMs. Moreover, we find that the lower the similarity between the candidate and the reference, the better the quality of the generated text. As shown in Figure 3, we select sentences that have various similarities

System	COMET-22↑	BLEU↑	COMET-22↑	BLEU↑	COMET-22↑	BLEU↑	COMET-22↑	BLEU↑
	DE-EN		ZH-EN		CS-EN		RU-EN	
WMT-Best†	85.0	33.4	81.0	**33.5**	**89.0**	**64.2**	86.0	**45.1**
MS-Translator†	84.7	33.5	80.4	27.9	87.4	54.9	85.2	43.9
GPT 1-shot	84.7	30.4	81.0	23.7	86.2	44.8	84.8	39.7
+ DTG	85.0	32.3	81.4	25.3	86.7	45.6	85.0	40.0
GPT 5-shot	85.3	32.3	81.1	23.6	86.9	47.2	84.9	39.9
+ DTG	85.4	33.2	81.7	25.2	87.0	47.4	85.1	40.3
GPT4 1-shot	85.6	33.5	82.4	26.0	87.3	48.1	86.1	43.1
+ DTG	**85.8**	**33.8**	**83.0**	26.4	87.7	49.4	**86.3**	43.7
	JA-EN		UK-EN		IS-EN		HA-EN	
WMT-Best†	81.6	24.8	**86.0**	**44.6**	**87.0**	**41.7**	**80.0**	**21.0**
MS-Translator†	81.5	24.5	83.5	42.4	85.9	40.5	73.3	16.2
GPT 1-shot	81.3	21.5	83.5	36.8	83.5	33.6	78.0	18.6
+ DTG	81.7	21.4	84.0	37.1	84.0	35.2	78.3	18.6
GPT 5-shot	81.2	20.5	84.0	38.0	84.1	35.0	78.3	18.8
+ DTG	82.2	22.4	84.2	39.0	84.6	36.0	78.6	19.2
GPT4 1-shot	83.4	24.7	85.7	39.9	86.9	39.9	77.5	18.3
+ DTG	**83.6**	**25.2**	85.9	40.6	**87.0**	40.9	77.9	18.9

Table 1: Evaluation results of GPT and GPT4 on six high-resource and two-low resource machine translation tasks from WMT Testsets. The best scores across different systems are marked in bold.

with the reference (using edit distance) as the synthesized sentence, and the performance decreases monotonically in general when the similarity increases. Therefore, we seek to choose a sentence that does not contain any correct information as the synthesized text. Potential candidates include a randomly sampled sentence or more extremely, an *empty string*, i.e., setting "[SYS]" as " ". Both choices successfully facilitate deliberation and consistently improve the outcomes across multiple text generation tasks. We use an empty string in our experiments as it is more general and elegant.

DTG has the following exceptional properties to steer LLMs on various text generation tasks:

- *Simple*: The final results can be obtained through a single-step inference of the LLM, without any additional resources or costs.
- *General*: It can be effortlessly applied to a broad range of text generation tasks only with minimal adjustments in the prompt.

4 Datasets and Evaluation

In experiments, we are devoted to evaluating the generation ability of LLMs and the proposed DTG prompting. We select 7 representative generation tasks, including machine translation, abstractive summarization, dialogue summarization, text simplification, style transfer, paraphrase and commonsense generation. Also, we expand the exploration to mathematical reasoning task, namely GSM8K.

We summarize the details of each dataset for each task, including the test sets, the selection of demonstrations (mostly from validation sets) and the corresponding prompts we have used. For more details please refer to the attached Appendix. Without meticulous parameter tuning, we set the *temperature* to 0 and *top_p* to 1 when calling the API.

5 Experiments

In this section, we assess the efficacy of the `text-davinci-003` (also known as GPT3.5, which is denoted as GPT in the following for simplicity) across 7 sequence generation tasks. The chosen baseline comparisons consist of 1-shot, and few-shot (mostly 5-shot) learning scenarios. To demonstrate the versatility of DTG method, we conduct further experiments with GPT4, a cutting-edge LLM API. Due to the considerable computational cost and API request constraints associated with the GPT4, it is challenging to perform extensive experiments. In the current manuscript, we only report the results on machine translation and text simplification.

5.1 Results on Machine Translation

We compare the performance of GPT standard prompting and our deliberate then generate method (DTG) with that of a commercial system (Microsoft Translator) in addition to WMT SoTA systems. Table 1 presents the results in both 1-shot and 5-shot scenarios. The findings here indicate that our reimplementation aligns with the trends observed in

System	CNN/DailyMail			GigaWord			SamSum			DialogSum		
	R1	R2	RL	R1	R2	RL	R1	R2	RL	R1	R2	RL
Transformer (Vaswani et al., 2017)	40.47	17.73	37.29	37.57	18.90	34.69	37.20	10.86	34.69	35.91	8.74	33.50
BART (Lewis et al., 2020)	**44.16**	**21.28**	**40.90**	**39.29**	**20.09**	**35.65**	**53.12**	**27.95**	**49.15**	**47.28**	**21.18**	44.83
UniLMv2 (Bao et al., 2020)	43.16	20.42	40.14	-	-	-	50.53	26.62	48.81	47.04	21.13	**45.04**
GPT 1-shot	38.87	15.36	35.11	31.24	11.61	27.99	44.52	19.92	39.60	36.84	14.23	32.20
+ DTG	40.17	15.60	36.04	31.50	12.00	28.50	45.50	20.58	40.13	39.01	15.50	34.13
GPT 5-shot	-	-	-	33.04	12.78	29.86	46.44	20.69	41.10	40.86	17.10	35.78
+ DTG	-	-	-	33.54	13.63	30.36	48.72	23.16	43.23	42.64	18.12	37.57
GPT 10-shot	-	-	-	33.24	13.26	30.46	47.37	22.08	42.20	41.28	17.48	36.69
+ DTG	-	-	-	34.02	14.21	31.04	50.48	24.88	45.31	45.11	19.50	39.71

Table 2: Experimental results on four summarization tasks.

System	GYAFC & EM		GYAFC & FR		Amazon		Yelp	
	BLEU	BLEURT	BLEU	BLEURT	BLEU	BLEURT	BLEU	BLEURT
Transformer†(Vaswani et al., 2017)	40.3	-	47.7	-	-	-	-	-
BART†(Lewis et al., 2020)	**76.9**	75.38	**79.3**	75.11	-	-	-	-
GPT 1-shot	52.9	73.42	44.6	70.73	36.1	64.56	30.9	64.03
+ DTG	66.8	75.20	65.9	74.60	35.4	63.60	31.3	64.19
GPT 5-shot	61.3	75.40	63.9	74.35	39.3	64.76	31.4	64.16
+ DTG	69.9	**76.36**	74.1	**75.43**	**40.9**	**65.42**	**32.2**	**64.87**

Table 3: Comparisons of 1-shot and 5-shot on four style transfer tasks, including Entertainment Music, Family Relationships, Amazon and Yelp. †denotes results borrowed from (Lai et al., 2021).

System	Asset	Wiki-auto
MUSS (Martin et al., 2022)	44.15	42.59
Control Prefix (Clive et al., 2022)	43.58	-
TST-Final (Omelianchuk et al., 2021)	41.46	-
GPT 1-shot	46.12	44.97
+ DTG	47.23	47.15
GPT 5-shot	45.95	45.12
+ DTG	47.05	47.54
GPT4 5-shot	47.10	45.96
+ DTG	**47.67**	**47.89**

Table 4: Comparisons of 1-shot, 5-shot with and without our DTG method on two text simplification tasks.

System	BLEU-3/4	ROUGE-2/L
BART (Lewis et al., 2020)	36.3/26.4	22.23/41.98
T5-Large (Raffel et al., 2020)	39.0/28.6	22.01/42.97
GPT 5-shot	39.7/30.0	25.28/46.55
+ DTG	**43.2/33.5**	**27.02/48.47**

Table 5: Results on the CommonGen benchmark.

System	Accuracy
GPT 8-shot	55.1
CoT 8-shot (Wei et al., 2022b)	59.8
+ DTG	64.5

Table 6: Results of GSM8K on DTG prompting.

previous study (Hendy et al., 2023), that 5-shot beats 1-shot in most language pairs. Benefiting from the deliberation, DTG effectively pushes the boundaries and leads to enhanced results across all to-English language pairs in both 1-shot and 5-shot settings based on GPT3.5 model. For instance, DTG method exhibits substantial BLEU score increases in DE-EN, ZH-EN, and UK-EN language pairs in 5-shot scenarios. More concretely, DTG even beats WMT-Best system in terms of COMET-22, which is a more recognized metric recently in the machine translation literature. Moreover, the consistent improvements on IS-EN and HA-EN demonstrate the effectiveness of DTG in low-resource settings.

Benefiting the strong comprehending ability of GPT4, we find there is no significant difference between 1-shot and 5-shot scenarios. Meanwhile, DTG is still effective on GPT4, showing consistent and indeed improvements in terms of COMET and BLEU. This finding demonstrates much stronger LLMs can still benefit from deliberation.

5.2 Results on Summarization

For abstractive summarization, we assess GPT models on CNN/DailyMail[1] and GigaWord, two benchmark datasets in the field. Additionally, we explore their efficacy in dialogue summarization, including SamSum and DialogSum[2], two hybrid tasks combining aspects of both dialogue and sum-

[1] Due to the limit of max length for GPT models (4097) and the long input length of CNN/DailyMail, we only evaluate the performance in 1-shot scenario.

[2] It is important to note that the results for DialogSum are averaged over three individual scores, each calculated using unique references spanning a range of topics.

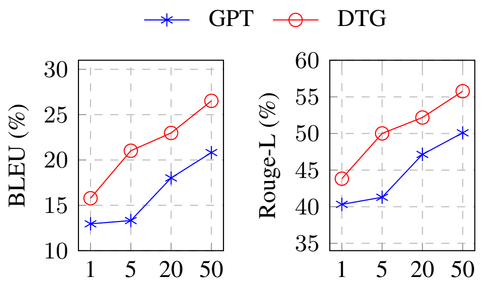

Figure 4: BLEU and ROUGE-L scores against the number of demonstrations on the paraphrase task.

#	System	BLEU	COMET
1	GPT 5-shot	23.6	81.12
2	+ DTG	25.2	81.70
3	+ w/o error detection	23.3	81.05
4	+ wrong error type	25.3	81.74
5	+ fixed error type	24.1	81.35
6	+ task-specific error type	25.5	81.77
7	+ fixed incorrect candidate	25.0	81.72
8	+ irrelevant languages	25.1	81.81
9	+ correct candidate	23.0	81.17

Table 7: Ablations on error types and candidae types.

marization. As shown in Table 2, GPT models show comparative performance with Transformer which is specially tuned on the downstream training set, e.g., Transformer. Our DTG delivers further improvements in terms of three ROUGE metrics, which demonstrate the effectiveness of DTG on long-term modeling task. Beyond this, DTG substantially incites GPT models to generate more precise summaries derived from extensive multi-turn dialogues. An upward trend in performance is observed with the introduction of additional demonstrations, further underscoring the effectiveness of the DTG method. However, DTG still lags behind of large-scale pretrained models, such as BART (Lewis et al., 2020) and UniLMv2 (Bao et al., 2020) in automatic evaluations. We will add more human-alignment judgment in Section 6.

5.3 Results on Style Transfer

Table 3 displays performance across style transfer tasks from the GYAFC dataset: Entertainment Music (EM) and Family Relationships (FR), both involving informal to formal transformations. Evidently, the Deliberate then Generate (DTG) method prompts the GPT model to correct inaccuracies and generate more precise informal sentences. Specifically, DTG achieves an 8-point and 10.04-point increase in BLEU score for EM and FR tasks, respectively, compared to standard prompting. Although DTG trails BART (Lewis et al., 2020) in BLEU scores, it surpasses BART in BLEURT scores, obtaining gains of 0.98 and 0.32 for EM and FR tasks, respectively. These results highlight the potential of LLMs and DTG method in style transfer tasks.

5.4 Results on Text Simplification

Experiments were conducted on two text simplification benchmarks, Asset and Wiki-Auto, where the primary goal is to create a simplified rendition of the given text input. The main evaluation metric is the SARI score. Our findings illustrate that GPT models demonstrate robust performance across both simplification benchmarks, even surpassing the existing state-of-the-art models (MUSS) built based on BART. Furthermore, the incorporation of DTG method significantly enhances GPT model performance, leading to improvements in both BLEU and SARI scores. Specifically, DTG establishes a new benchmark for state-of-the-art results on these two simplification tasks.

5.5 Results on Commonsense Generation

Table 5 summarizes the comparison between GPT models with and without DTG method on an open Commonsense generation benchmark. This task is more flexible than the aforementioned, meanwhile raising the evaluation difficulty. We see that GPT models with standard prompting even surpass large-scale pretrained generation models, such as BART (Lewis et al., 2019) and T5 (Raffel et al., 2020). DTG achieves further improvements in terms of BLEU-3/BLEU-4 and ROUGE-2/ROUGE-L, resulting in an average of 3.50 BLEU and almost 2.00 ROUGE improvements. This also establishes a new SoTA on this benchmark.

5.6 Results on Paraphrase

Figure 4 plots the BLEU and ROUGE-L scores for GPT and DTG in relation to various few-shot scenarios. We find that DTG outperforms GPT models in terms of both BLEU and ROUGE-L metrics across all scenarios. However, only 5-shot demonstrations cannot enable LLMs to clearly capture the underlying mapping rule between the source and the the target. Interestingly, a significant enhancement in DTG performance is observed with the increase in the number of demonstrations. This improvement can be attributed to the model's enhanced ability to comprehend the underlying mapping rules with the expanded demonstration set.

System	ZH-EN		Asset	
	BLEU	Human	SARI	Human
GPT 5-shot	22.8	4.16	45.95	11.6%
DTG 5-shot	24.9	**4.39**	47.05	**67.4%**

Table 8: Human evaluation on DTG prompting.

5.7 Results on Mathematical Reasoning

While our primary focus is on evaluating LLMs for text generation, we extend our analysis to reasoning tasks, such as GSM8K (Cobbe et al., 2021). Table 6 compares the accuracy of standard prompting, CoT, and DTG. Our results show that DTG, when combined with CoT, achieves an accuracy of 64.5 in 8-shot scenarios, indicating its utility beyond text generation.

6 Analysis

In this section, we delve into a series of intriguing questions to elucidate the circumstances and reasons underpinning the robust performance of DTG. Unless specified otherwise, the base engine utilized throughout this investigation is `text-davinci-003`.

Ablations on Error Types Prior research underscores the significant impact of both the quality and quantity of demonstrations (Zhang et al., 2023; Vilar et al., 2022; Agrawal et al., 2022) . Thus, we would like to discern whether the improvements are attributable to template modifications or the deliberate capability inherent to the LLMs. Table 7 summarizes the comparisons on WMT ZH-EN. Firstly, DTG experiences a significant degradation in BLEU score when removing the explicitly error detection prompt[3], suggesting that the excised segment may contain crucial triggers stimulating the deliberate capability of the LLM. Along this line, by comparing #4[4], #5[5] and # 6 with #2, we can conclude 1) LLMs can rethink by themselves and make "correct" decisions though the demonstration is incorrect. 2) Restricting the thought of LLMs would hinder the performance. 3) Adding task-specific error type results in better generation.

Ablations on Candidates Here, we aim to explore if other candidates rather than *empty string* may also prove effective in DTG. The last two lines in Table 7 shows the comparison. Specifically, the term "fixed incorrect candidate" (#7) refers to the use of a fixed yet incorrect (irrelevant) English translation as the candidate.[6] Likewise, system #8 indicates that the candidates neither belong to the target language nor conform to the correct structure or grammar.[7] Interestingly, both 2 systems deliver comparable performance with our default setting, with system #8 even achieving a higher COMET score. However, when shifting to a correct candidate, LLMs seem to underperform. This observation suggests that LLMs can effectively deliberate when the candidate is incorrect - whether it is an *empty string* or other incorrect translations - and subsequently generate a substantially improved translation.

Evaluation by GPT Models As previously discussed, despite DTG's impressive performance, it falls short of BART in some scenarios—most notably, it exhibits a significant gap in terms of ROUGE scores in summarization tasks. However, Liu et al. (2023) suggested that ROUGE may not accurately represent the true performance of summarization tasks, given its poor alignment with human evaluations. In contrast, GPT models achieve optimal alignment with human justification and substantially outperform all previous state-of-the-art evaluators on the SummEval benchmark. This observation prompts an investigation into whether the generation output by DTG can surpass that of BART. Following their suggestion, we conduct reference-based evaluation and design a prompt as shown in Figure 10. We extract 500 test sets and compared DTG with the best result using GPT3.5 and GPT4 to select a better candidate. Results in Figure 5 reveal that DTG significantly beats the best system within GPT evaluation.

Human Evaluation We further conducted human evaluation with human assessments on translation (randomly selected 500 cases) and simplification tasks to mitigate potential bias in GPT models favoring their own outputs. Annotators scored ZH-EN translations on a 1-5 scale and indicated preferences for the Asset task. It's worth noting that for the Asset task, some cases showed no sig-

[3] eliminating the phrase "Please detect the error type firstly, and refine the translation then"

[4] replacing "incorrect translation" with "good/correct translation" in the demonstration only

[5] replacing "incorrect translation" with "good/correct translation" in the demonstration only

[6] We random sample an English sentence: [SYS]: *EBA Education Team together with Accace Ukraine invite you to join the EBA Education Update: Performance Audit.*

[7] Similarly, we random sample an Ukraine sentence: [SYS]: З впевненістю можете довіряти нам і будь ласка, звертайтеся до нас, якщо у вас є які-небудь питання чи коментарі.

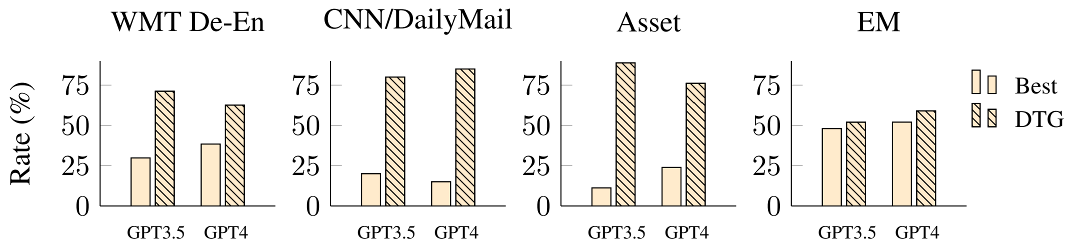

Figure 5: GPT3.5 and GPT4 evaluation on 4 generation tasks. Note that we random select 500 samples due to the limitation of GPT4 access.

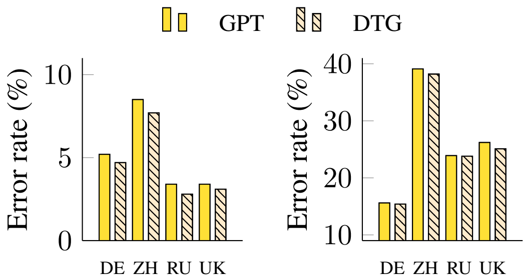

Figure 6: Statistics of error rate for under translation (above) and entity translation (below).

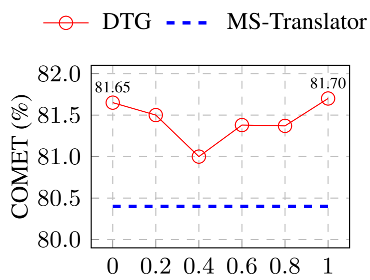

Figure 7: COMET *v.s.* word drop rate of MS-Translator candidate.

nificant difference in performance between the two methods (neutral). Detailed scoring guidelines are provided in the Appendix. As shown in Table 8, DTG outperforms the standard prompt in human evaluations across both tasks.

Error Statistical Analysis To evaluate whether the proposed DTG prompting can facilitate error avoidance in GPT, we conduct error statistics on machine translation, where two frequently occurring error types are considered (i.e., under translation and incorrect entity translation) (Hassan et al., 2018). Figure 6 provides a comparison of the error rates between GPT models with and without the application of the DTG method. It is obvious to see that DTG reduces both error rates compared with the direct generation manner.

DTG Can Serve as A Good Refiner To explore how DTG performs relative to the quality of input candidates, we experimented on the ZH-EN translation task. Candidates of varying quality were generated by selectively omitting words from MS-Translator outputs. Figure 7 plots COMET scores against word drop rates, comparing MS-Translator (blue line) with DTG-augmented candidates (red line). GPT improves MS-Translator's COMET score from 80.4 to 81.65. While DTG underperforms when refining its own correct translations (see Table 7), it excels with candidates from other systems. This confirms DTG's versatility in improving even high-quality candidates. Interestingly, DTG's performance dips with increased word omissions but improves when the candidate is nearly an *empty string*. Using an *empty string* as a candidate offers a resource-efficient way to boost performance without specialized demonstration crafting.

7 Conclusions

In this paper, we propose DTG prompting, which encourages LLMs to deliberate before generating the final results by letting the model detect the error type on a synthetic text that may contain errors. Using an empty string as the synthetic text successfully gets rid of an extra baseline system and improves the quality of the generated text. The DTG prompting can be easily applied to various text generation tasks with minimal adjustments in the prompt. Extensive experiments conducted on over 20 datasets across 7 text generation tasks demonstrate the effectiveness and broad applicability of the DTG prompting. One potential avenue for further enhancing the efficacy of DTG prompting involves leveraging task-specific domain knowledge. (e.g., explicitly listing the potential error types in the prompts to provide guidance for deliberation), which is worth future investigation.

Limitation

Due to restricted access to GPT4, we have evaluated our *Deliberate then Generate* (DTG) method on just two generation tasks: machine translation (across 8 language pairs) and simplification. There exists a necessity for more expansive experimentation across other tasks. Additionally, the effectiveness of DTG is contingent on model capacity. Models such as LLaMa-7B might not fully comprehend the instructions provided, resulting in weaker performance on downstream tasks. In our future work, we aim to ascertain the required scale of a language model to successfully facilitate deliberative generation.

Our work inherits the biases from pre-trained language models. For example, we only conduct experiments on English generation that GPT models are most powerful at. We provide results and analysis on English-to-Others translation in Appendix D. Future works could investigate the performance of DTG on multilingual pre-trained models.

Ethical Statement

All experiments in our work were conducted on existing datasets commonly employed in prior publicly available research publications. We keep fair and honest in our analysis of experimental results, and our work does not harm anyone. Additionally, we will make our code accessible for future investigations.

References

Sweta Agrawal, Chunting Zhou, Mike Lewis, Luke Zettlemoyer, and Marjan Ghazvininejad. 2022. In-context examples selection for machine translation. *arXiv preprint arXiv:2212.02437*.

Fernando Alva-Manchego, Louis Martin, Antoine Bordes, Carolina Scarton, Benoît Sagot, and Lucia Specia. 2020. ASSET: A dataset for tuning and evaluation of sentence simplification models with multiple rewriting transformations. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 4668–4679, Online. Association for Computational Linguistics.

Hangbo Bao, Li Dong, Furu Wei, Wenhui Wang, Nan Yang, Xiaodong Liu, Yu Wang, Jianfeng Gao, Songhao Piao, Ming Zhou, and Hsiao-Wuen Hon. 2020. Unilmv2: Pseudo-masked language models for unified language model pre-training. In *Proceedings of the 37th International Conference on Machine Learning, ICML 2020, 13-18 July 2020, Virtual Event*, volume 119 of *Proceedings of Machine Learning Research*, pages 642–652. PMLR.

Tom Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared D Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, et al. 2020. Language models are few-shot learners. *Advances in neural information processing systems*, 33:1877–1901.

Yulong Chen, Yang Liu, Liang Chen, and Yue Zhang. 2021. DialogSum: A real-life scenario dialogue summarization dataset. In *Findings of the Association for Computational Linguistics: ACL-IJCNLP 2021*, pages 5062–5074, Online. Association for Computational Linguistics.

Jordan Clive, Kris Cao, and Marek Rei. 2022. Control prefixes for parameter-efficient text generation. In *Proceedings of the 2nd Workshop on Natural Language Generation, Evaluation, and Metrics (GEM)*, pages 363–382, Abu Dhabi, United Arab Emirates (Hybrid). Association for Computational Linguistics.

Karl Cobbe, Vineet Kosaraju, Mohammad Bavarian, Mark Chen, Heewoo Jun, Lukasz Kaiser, Matthias Plappert, Jerry Tworek, Jacob Hilton, Reiichiro Nakano, Christopher Hesse, and John Schulman. 2021. Training verifiers to solve math word problems. *arXiv preprint arXiv:2110.14168*.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2018. Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805*.

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

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

Bogdan Gliwa, Iwona Mochol, Maciej Biesek, and Aleksander Wawer. 2019. SAMSum corpus: A human-annotated dialogue dataset for abstractive summarization. In *Proceedings of the 2nd Workshop on New Frontiers in Summarization*, pages 70–79, Hong Kong, China. Association for Computational Linguistics.

Shansan Gong, Mukai Li, Jiangtao Feng, Zhiyong Wu, and LingPeng Kong. 2022. Diffuseq: Sequence to sequence text generation with diffusion models. *arXiv preprint arXiv:2210.08933*.

Hany Hassan, Anthony Aue, Chang Chen, Vishal Chowdhary, Jonathan Clark, Christian Federmann, Xuedong Huang, Marcin Junczys-Dowmunt, William Lewis, Mu Li, et al. 2018. Achieving human parity on automatic chinese to english news translation. *arXiv preprint arXiv:1803.05567*.

Amr Hendy, Mohamed Abdelrehim, Amr Sharaf, Vikas Raunak, Mohamed Gabr, Hitokazu Matsushita, Young Jin Kim, Mohamed Afify, and Hany Hassan

Awadalla. 2023. How good are gpt models at machine translation? a comprehensive evaluation. *arXiv preprint arXiv:2302.09210*.

Edward J Hu, Yelong Shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, and Weizhu Chen. 2021. Lora: Low-rank adaptation of large language models. *arXiv preprint arXiv:2106.09685*.

Chao Jiang, Mounica Maddela, Wuwei Lan, Yang Zhong, and Wei Xu. 2020. Neural CRF model for sentence alignment in text simplification. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 7943–7960, Online. Association for Computational Linguistics.

Wenxiang Jiao, Jen tse Huang, Wenxuan Wang, Zhiwei He, Tian Liang, Xing Wang, Shuming Shi, and Zhaopeng Tu. 2023. Parrot: Translating during chat using large language models tuned with human translation and feedback. In *Findings of EMNLP*.

Takeshi Kojima, Shixiang Shane Gu, Machel Reid, Yutaka Matsuo, and Yusuke Iwasawa. 2022. Large language models are zero-shot reasoners. *arXiv preprint arXiv:2205.11916*.

Huiyuan Lai, Antonio Toral, and Malvina Nissim. 2021. Thank you BART! rewarding pre-trained models improves formality style transfer. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 2: Short Papers)*, pages 484–494, Online. Association for Computational Linguistics.

Mike Lewis, Yinhan Liu, Naman Goyal, Marjan Ghazvininejad, Abdelrahman Mohamed, Omer Levy, Ves Stoyanov, and Luke Zettlemoyer. 2019. Bart: Denoising sequence-to-sequence pre-training for natural language generation, translation, and comprehension. *arXiv preprint arXiv:1910.13461*.

Mike Lewis, Yinhan Liu, Naman Goyal, Marjan Ghazvininejad, Abdelrahman Mohamed, Omer Levy, Veselin Stoyanov, and Luke Zettlemoyer. 2020. BART: Denoising sequence-to-sequence pre-training for natural language generation, translation, and comprehension. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 7871–7880, Online. Association for Computational Linguistics.

Xiang Lisa Li and Percy Liang. 2021. Prefix-tuning: Optimizing continuous prompts for generation. *arXiv preprint arXiv:2101.00190*.

Bill Yuchen Lin, Wangchunshu Zhou, Ming Shen, Pei Zhou, Chandra Bhagavatula, Yejin Choi, and Xiang Ren. 2020. CommonGen: A constrained text generation challenge for generative commonsense reasoning. In *Findings of the Association for Computational Linguistics: EMNLP 2020*, pages 1823–1840, Online. Association for Computational Linguistics.

Chin-Yew Lin. 2004. ROUGE: A package for automatic evaluation of summaries. In *Text Summarization Branches Out*, pages 74–81, Barcelona, Spain. Association for Computational Linguistics.

Yang Liu, Dan Iter, Yichong Xu, Shuohang Wang, Ruochen Xu, and Chenguang Zhu. 2023. Gpteval: Nlg evaluation using gpt-4 with better human alignment. *arXiv preprint arXiv:2303.16634*.

Gary F Marcus. 1993. Negative evidence in language acquisition. *Cognition*, 46(1):53–85.

Louis Martin, Angela Fan, Éric de la Clergerie, Antoine Bordes, and Benoît Sagot. 2022. MUSS: Multilingual unsupervised sentence simplification by mining paraphrases. In *Proceedings of the Thirteenth Language Resources and Evaluation Conference*, pages 1651–1664, Marseille, France. European Language Resources Association.

Harsha Nori, Nicholas King, Scott Mayer McKinney, Dean Carignan, and Eric Horvitz. 2023. Capabilities of gpt-4 on medical challenge problems. *arXiv preprint arXiv:2303.13375*.

Kostiantyn Omelianchuk, Vipul Raheja, and Oleksandr Skurzhanskyi. 2021. Text Simplification by Tagging. In *Proceedings of the 16th Workshop on Innovative Use of NLP for Building Educational Applications*, pages 11–25, Online. Association for Computational Linguistics.

OpenAI. 2023. Gpt-4 technical report. *arXiv preprint arXiv:2303.08774*.

Long Ouyang, Jeffrey Wu, Xu Jiang, Diogo Almeida, Carroll Wainwright, Pamela Mishkin, Chong Zhang, Sandhini Agarwal, Katarina Slama, Alex Ray, et al. 2022. Training language models to follow instructions with human feedback. *Advances in Neural Information Processing Systems*, 35:27730–27744.

Keqin Peng, Liang Ding, Qihuang Zhong, Li Shen, Xuebo Liu, Min Zhang, Yuanxin Ouyang, and Dacheng Tao. 2023. Towards making the most of chatgpt for machine translation. *arXiv preprint arXiv:2303.13780*.

Matt Post. 2018. A call for clarity in reporting BLEU scores. In *Proceedings of the Third Conference on Machine Translation: Research Papers*, pages 186–191, Belgium, Brussels. Association for Computational Linguistics.

Alec Radford, Jeffrey Wu, Rewon Child, David Luan, Dario Amodei, Ilya Sutskever, et al. 2019. Language models are unsupervised multitask learners. *OpenAI blog*, 1(8):9.

Colin Raffel, Noam Shazeer, Adam Roberts, Katherine Lee, Sharan Narang, Michael Matena, Yanqi Zhou, Wei Li, and Peter J. Liu. 2020. Exploring the limits of transfer learning with a unified text-to-text transformer. *J. Mach. Learn. Res.*, 21:140:1–140:67.

Sudha Rao and Joel Tetreault. 2018. Dear sir or madam, may I introduce the GYAFC dataset: Corpus, benchmarks and metrics for formality style transfer. 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 129–140, New Orleans, Louisiana. Association for Computational Linguistics.

Ricardo Rei, José G. C. de Souza, Duarte Alves, Chrysoula Zerva, Ana C Farinha, Taisiya Glushkova, Alon Lavie, Luisa Coheur, and André F. T. Martins. 2022. COMET-22: Unbabel-IST 2022 submission for the metrics shared task. In *Proceedings of the Seventh Conference on Machine Translation (WMT)*, pages 578–585, Abu Dhabi, United Arab Emirates (Hybrid). Association for Computational Linguistics.

Victor Sanh, Albert Webson, Colin Raffel, Stephen H Bach, Lintang Sutawika, Zaid Alyafeai, Antoine Chaffin, Arnaud Stiegler, Teven Le Scao, Arun Raja, et al. 2021. Multitask prompted training enables zero-shot task generalization. *arXiv preprint arXiv:2110.08207*.

Timo Schick, Jane Dwivedi-Yu, Roberto Dessì, Roberta Raileanu, Maria Lomeli, Luke Zettlemoyer, Nicola Cancedda, and Thomas Scialom. 2023. Toolformer: Language models can teach themselves to use tools. *arXiv preprint arXiv:2302.04761*.

Timo Schick and Hinrich Schütze. 2020. Exploiting cloze questions for few shot text classification and natural language inference. *arXiv preprint arXiv:2001.07676*.

Thibault Sellam, Dipanjan Das, and Ankur P Parikh. 2020. Bleurt: Learning robust metrics for text generation. In *Proceedings of ACL*.

Yongliang Shen, Kaitao Song, Xu Tan, Dongsheng Li, Weiming Lu, and Yueting Zhuang. 2023. Hugginggpt: Solving ai tasks with chatgpt and its friends in huggingface. *arXiv preprint arXiv:2303.17580*.

Hugo Touvron, Thibaut Lavril, Gautier Izacard, Xavier Martinet, Marie-Anne Lachaux, Timothée Lacroix, Baptiste Rozière, Naman Goyal, Eric Hambro, Faisal Azhar, et al. 2023. Llama: Open and efficient foundation language models. *arXiv preprint arXiv:2302.13971*.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Lukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. In *Proc. of NeurIPS*, pages 5998–6008.

David Vilar, Markus Freitag, Colin Cherry, Jiaming Luo, Viresh Ratnakar, and George Foster. 2022. Prompting palm for translation: Assessing strategies and performance. *arXiv preprint arXiv:2211.09102*.

Jason Wei, Maarten Bosma, Vincent Y Zhao, Kelvin Guu, Adams Wei Yu, Brian Lester, Nan Du, Andrew M Dai, and Quoc V Le. 2021. Finetuned language models are zero-shot learners. *arXiv preprint arXiv:2109.01652*.

Jason Wei, Xuezhi Wang, Dale Schuurmans, Maarten Bosma, Ed Chi, Quoc Le, and Denny Zhou. 2022a. Chain of thought prompting elicits reasoning in large language models. *arXiv preprint arXiv:2201.11903*.

Jason Wei, Xuezhi Wang, Dale Schuurmans, Maarten Bosma, Brian Ichter, Fei Xia, Ed H. Chi, Quoc V. Le, and Denny Zhou. 2022b. Chain-of-thought prompting elicits reasoning in large language models. In *NeurIPS*.

Chenfei Wu, Shengming Yin, Weizhen Qi, Xiaodong Wang, Zecheng Tang, and Nan Duan. 2023. Visual chatgpt: Talking, drawing and editing with visual foundation models. *arXiv preprint arXiv:2303.04671*.

Wei Xu, Courtney Napoles, Ellie Pavlick, Quanze Chen, and Chris Callison-Burch. 2016. Optimizing statistical machine translation for text simplification. *Transactions of the Association for Computational Linguistics*, 4:401–415.

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

Chuanyang Zheng, Zhengying Liu, Enze Xie, Zhenguo Li, and Yu Li. 2023. Progressive-hint prompting improves reasoning in large language models. *arXiv preprint arXiv:2304.09797*.

Yongchao Zhou, Andrei Ioan Muresanu, Ziwen Han, Keiran Paster, Silviu Pitis, Harris Chan, and Jimmy Ba. 2022. Large language models are human-level prompt engineers. *arXiv preprint arXiv:2211.01910*.

A Datasets and Evaluation

In experiments, we are devoted to evaluating the generation ability of LLMs and the proposed DTG prompting. We select 7 representative generation tasks, including machine translation, abstractive summarization, dialogue summarization, text simplification, style transfer, paraphrase and commonsense generation.

Machine Translation For the machine translation task, we aligned with Hendy et al. (2023)'s work and experimented on both high-resource and low-resource scenarios. For the high-resource setting, we include German, Czech, Chinese, Japanese, Russian, and Ukrainian paired with English. In the low-resource context, we examine Icelandic and Hausa. The performance is evaluated in terms of SacreBLEU[8] (Post, 2018), ChrF, TER (translation error rate) and COMET-22 (Rei et al., 2022).

Abstractive Summarization We also evaluate LLM's ability to process long sequence on CNN-DailyMail and Gigaword, two widely used abstractive summarization datasets. The evaluation metric is F1-ROUGE (Lin, 2004), consisting of ROUGE-1, ROUGE-2 and ROUGE-L.

Dialog Summarization Dialogue summarization presents greater challenges than traditional text summarization due to the intricate conversation contexts that models need to comprehend, though their contexts are relatively shorter. This attribute enables us to test few-shot abilities due to the restricted input length. To investigate this, we select SamSum[9] (Gliwa et al., 2019) and DialogSum[10] (Chen et al., 2021), two benchmark datasets for dialogue summarization. The evaluation metric is the same as abstractive summarization.

Text Simplification The purpose of text simplification is to revise complex text into sequences with simplified grammar and word choice. In this work, we mainly report the performance on two benchmarks, namely Asset (Alva-Manchego et al., 2020) and Wiki-auto (Jiang et al., 2020). Asset is a multi-reference dataset for the evaluation of sentence simplification in English. The dataset uses the same 2,359 sentences from TurkCorpus (Xu et al., 2016) and each sentence is associated with 10 crowdsourced simplifications. Similarly, each test set in Wiki-auto owns 8 references. We use SacreBLEU and BLEURT as the metric.

Style Transfer We used three widely-used English transfer learning datasets, namely Grammarly's Yahoo Answers Formality Corpus (GYAFC), Amazon and Yelp reviews. The GYAFC dataset (Rao and Tetreault, 2018) was originally a question-and-answer dataset on an online forum, consisting of informal and formal sentences from the two categories: Entertainment & Music (EM) and Family & Relationships (FR). Both FR and EM provide 4 references to evaluate the fidelity. The Amazon dataset is a product review dataset, labeled as either a positive or negative sentiment. Similarly, the Yelp dataset is a restaurant and business review dataset with positive and negative sentiments. Both Amazon and Yelp are single-reference. The evaluation metrics contain BLEU and BLEURT (Sellam et al., 2020).

Paraphrase We endeavor to evaluate the paraphrase ability of LLMs upon the well-known Quora Question Pairs (QQP) dataset, which requires generating an alternative surface form in the same language expressing the same semantic content. We utilize the preprocessed data from (Gong et al., 2022). The evaluation metrics covers BLEU and ROUGE-L for a comprehensive comparison.

Common Sense Generation We choose CommonGen (Lin et al., 2020), a novel constrained generation task that requires models to generate a coherent sentence with the providing key concepts. We report both BLEU-3/4 and ROUGE-2/L to keep a fair comparison with results in prior work (Lin et al., 2020).

Reasoning For the reasoning task, we evaluate our method on a widely used benchmark, GSM8K (Cobbe et al., 2021), a challenging dataset consisting of high-quality linguistically diverse grade school math word problems. We report the accuracy of the 8-shot demonstration on the test set including 1,319 mathematical questions.

B Details of Datasets

In this section, we offer more detailed statistics concerning the test sets utilized in this study, encompassing 8 machine translation, 4 summarization,

[8] BLEU+case.mixed+numrefs.1+smooth.exp+tok.13a +version.2.3.1

[9] https://huggingface.co/datasets/samsum

[10] https://github.com/cylnlp/DialogSum

Dataset	Num.	Total Words	Ave. Words	Dataset	Num.	Total Words	Ave. Words
WMT DE-EN	1984	33540	16.9	CNN/DailyMail	11490	9017116	784.8
WMT CS-EN	1448	26050	17.9	GigaWord	1951	72171	37.0
WMT JA-EN	2008	36731	18.3	SamSum	819	104492	127.6
WMT ZH-EN	1875	14353	7.7	DialogSum	500	96385	192.7
WMT RU-EN	2016	32992	16.3	EM	1416	17279	12.2
WMT UK-EN	2018	29273	14.5	FR	1332	16799	12.6
WMT IS-EN	1000	19930	19.9	Amazon	500	6055	12.1
WMT HA-EN	997	30955	31.0	Yelp	500	5432	10.9
CommonGen	1497	6465	6.5	Asset	359	8115	22.6
QQP	2500	27543	11.0	Wiki-auto	2000	43860	21.9

Table 9: Statistics of the dataset we used on over 20 benchmarks. Note that “Num.” represents the number of test sets for each benchmark. “Total Words” and “Ave. Words” denote the total word count and average lengths, respectively. These statistics are based on tokenization sequences.

System	Score2	Score3	Score4	Score5
GPT 5-shot	16	88	196	200
DTG 5-shot	5	45	200	250

Table 10: Detailed score distribution of human evaluation on ZH-EN.

4 style transfer, 2 simplification, 1 commonsense generation, and 1 paraphrase benchmarks. Table 9 provides a summary of the number of test sets, total words, and the average length. We will release the test sets and the corresponding demonstrations in the future. Note that the statistic is conducted based on tokenization sequences, which would be further segmented by BPE before feeding into LLMs. Consequently, the average length of summarization inputs would appear significantly larger, leading to an elevated risk in the context of few-shot requests.

C Design of Prompts

Figure 8 presents the DTG demonstration design across the other three text generation tasks. It can be observed that DTG does not necessitate task-specific designs; instead, a clear instruction outlining the main task for each work suffices. For the ease of replication of our results, we also furnish all baseline prompts, as depicted in Figure 9. Also, we provide the prompting design for GPT evaluation in Figure 10, which follows a zero-shot fashion.

To facilitate a more comprehensive understanding of the prompt ablations conducted in Section 6, we provide the corresponding design of prompts in Figure 11. Please note that prompts in blue represent the pre-designed demonstration, while those in red represent the test input. As observed, firstly, removing the error detection leads to the prompting in 11 (a). Additionally, the term “wrong error type” implies that we fed an *empty string* into LLMs, presenting it as a good translation. However, LLMs can autonomously detect the correct error type as an “incorrect translation” and subsequently generate an accurate response following careful deliberation (Figure 11 (b)). Conversely, if we constrain the error type detection process and solely allow LLMs to generate the translation, a considerable performance gap emerges (See Figure 11 (c)).

D More Analyses

Results on Machine Translation from English Table 11 summarizes the results of standard prompting and our DTG method in 5-shot scenarios, alongside results from WMT-Best and MS-Translator. When compared to results from to-English directional language pairs, such as DE-EN, the improvements provided by DTG over the standard prompting strategy appear somewhat marginal. Furthermore, DTG may yield results inferior to standard prompting in EN-ZH and EN-UK scenarios. This can likely be ascribed to the disparities in the balance of training sets across different languages.

Details for Human Evaluation We have further conducted human evaluations to obtain more convincing results. Given the constraints of human effort, we have focused our evaluation solely on ZH-EN translation and Asset simplification. It’s important to note that, specifically for the ZH-EN translation, we have devised the following rules for human evaluators:

- 1 point - No translation or only isolated words translated.
- 2 points - 50% errors in translation; meaning distorted.

Dialogue Summarization

Given the English dialogue: [SRC]

the already generated dialogue summarization is: [SYS]

Please detect the error type firstly, and provide the refined summarization then.

Error type: incorrect summarization, the refined summarization is: [TGT]

...

Paraphrase

Given the English sentence: [SRC]

the already generated paraphrase is: [SYS]

Please detect the error type firstly, and provide the refined paraphrase then.

Error type: incorrect paraphrase, the refined paraphrase is: [TGT]

...

Commonsense Generation

Given several key words: [SRC]

the already generated sentence using background commonsense knowledge is: [SYS]

Please detect the error type firstly, and provide the refined sentence then.

Error type: incorrect generation, the refined sentence is: [TGT]

...

Figure 8: Illustration of DTG demonstration design for dialogue summarization, paraphrase and commonsense generation tasks within minimal modifications.

Translation

Given the [src] sentence: [SRC]

the [tgt] translation of the sentence is: [TGT]

Given the [src] sentence: [Input]

the [tgt] translation of the sentence is:

Dialogue Summarization

Given the English dialogue: [SRC]

please summarize the main context: [TGT]

Given the English dialogue: [Input]

please summarize the main context:

Simplification

Given the English sentence: [SRC]

the simplification of the sentence is: [TGT]

Given the English sentence: [Input]

the simplification of the sentence is:

Style Transfer

Given the English sentence: [SRC]

please transfer the style of the sentence into formal: [TGT]

Given the English sentence: [Input]

please transfer the style of the sentence into formal:

Paraphrase

Given the English sentence: [SRC]

the paraphrase of the sentence is: [TGT]

Given the English sentence: [Input]

the paraphrase of the sentence is:

Commonsense Generation

Given several key words: [SRC]

Please generate a coherent sentence using background commonsense knowledge with the providing key words: [TGT]

Given several key words: [Input]

Please generate a coherent sentence using background commonsense knowledge with the providing key words:

Figure 9: Illustration of the standard GPT prompting involving both demonstration and test input on six generation tasks, including machine translation, dialogue summarization, text simplification, style transfer, paraphrase and commonsense generation.

Prompt template of GPT evaluation

Test

Given the [src] sentence: [SRC]

Your task is to score the following two candidates translated by two systems, Candidate1: [sys1] Candidate2: [sys2].

Please select the better one in terms of both coherence and fidelity. Note that C1 for Candidate1, C2 for Candidate2.

Output:

Figure 10: Illustration of the prompting design of GPT evaluation for Figure 5. We adhere to the recommendation proposed in (Liu et al., 2023)'s work, implementing a zero-shot GPT evaluation approach to identifying superior candidate translations through the adjudication of LLMs.

(a) Prompt template of "w/o error detection"

Demonstration

```
Given the [src] sentence: [SRC]
the [tgt] translation is: [SYS]
The refined [tgt] translation is: [TGT]
```

Test

```
Given the [src] sentence: [SRC]
the [tgt] translation is: [SYS]
The refined [tgt] translation is:
```

(b) Prompt template of "wrong error type"

Demonstration

```
Given the [src] sentence: [SRC]
the [tgt] translation is: [SYS]
Please detect the error type firstly, and refine the translation then.
Error type: good/correct translation, the refined [tgt] translation is: [TGT]
```

Test

```
Given the [src] sentence: [SRC]
the [tgt] translation is: [SYS]
Please detect the error type firstly, and refine the translation then.
Error type:
```

(c) Prompt template of "fixed error type"

Demonstration

```
Given the [src] sentence: [SRC]
the [tgt] translation is: [SYS]
Please detect the error type firstly, and refine the translation then.
Error type: under translation, the refined [tgt] translation is: [TGT]
```

Test

```
Given the [src] sentence: [SRC]
the [tgt] translation is: [SYS]
Please detect the error type firstly, and refine the translation then.
Error type: under translation, the refined [tgt] translation is:
```

Figure 11: Illustration of the prompting design of the ablation study in Table 7. Note that all [SYS] here is *empty string*. The purpose here is to evaluate the deliberation ability of LLMs.

Table 11: Evaluation results of GPT on six high-resource and two-low resource machine translation tasks from WMT Testsets in from English directions. The best scores are marked in bold.

System	COMET-22↑	TER↓	ChrF↑	BLEU↑	COMET-22↑	TER↓	ChrF↑	BLEU↑
	EN-DE				EN-ZH			
WMT-Best†	**87.2**	**49.9**	**64.6**	**38.4**	**86.7**	102.3	41.1	44.8
MS-Translator†	86.8	50.5	64.2	37.3	86.1	**94.2**	**43.1**	**48.1**
GPT 5-shot	86.3	54.6	61.3	33.3	**86.7**	97.4	40.0	43.7
+ DTG	86.3	54.1	61.6	33.4	86.6	98.6	39.4	43.5
	EN-CS				EN-RU			
WMT-Best†	**91.9**	**43.7**	**68.2**	**45.8**	**89.5**	56.8	**58.3**	32.4
MS-Translator†	90.6	45.7	65.6	42.1	87.4	**56.7**	58.1	**33.1**
GPT 5-shot	88.9	54.6	58.9	32.7	87.0	61.3	54.4	28.2
+ DTG	88.8	54.5	59.0	32.9	85.7	63.0	52.1	28.1
	EN-JA				EN-UK			
WMT-Best†	**89.3**	**105.9**	**36.8**	**27.6**	**88.8**	**57.5**	**59.3**	**32.5**
MS-Translator†	88.0	106.0	34.9	25.1	86.1	63.2	56.1	28.2
GPT 5-shot	88.1	111.8	31.0	21.4	85.4	70.2	50.6	21.8
+ DTG	88.0	111.8	31.0	21.7	83.8	71.6	47.8	20.8
	EN-IS				EN-HA			
WMT-Best†	**86.8**	**55.0**	**59.6**	**33.3**	**79.8**	**65.6**	**51.1**	**20.1**
MS-Translator†	84.3	57.2	56.8	28.7	72.5	75.6	38.4	10.3
GPT 5-shot	76.1	70.8	44.1	16.2	72.8	87.4	38.5	9.9
+ DTG	76.7	70.9	44.2	16.3	73.2	77.7	39.3	10.1

Source	味道赞，肉类好，服务热情
Reference	Nice taste, great meat, enthusiastic service.
GPT 1-shot	The taste is great, the meat is good, and the service is enthusiastic.
+ Refine	The flavors are amazing, the meat is excellent, and the service is warm and welcoming.
+ DTG	Great taste, good meat, enthusiastic service.
Source	目前已经购买了这个系列3款机器！
Reference	I have bought three laptops of this series!
GPT 1-shot	So far, 3 machines from this series have been purchased!
+ Refine	Up until now, 3 machines from this series have been purchased!
+ DTG	I have already purchased 3 models from this series!

Table 12: Case study on refining from the previous candidate (Refine) and the proposed DTG method.

- 3 points - Mostly accurate; minor errors and inconsistencies.
- 4 points - Generally correct; some language and spacing issues.
- 5 points - Smooth, accurate, and fully conveys the original meaning.

Note that the 500 sentences were randomly selected from the test set. We also provide the detailed score distribution:

Case Study We provide a case study based on GPT4 model in Table 12, where "Refine" indicates utilizing the 5-shot baseline results as the synthesized sentences, i.e., "[INCORRECT SYS]" in Figure 1, and DTG is our method that uses an empty string instead. The conclusions are two-fold. 1) Using the baseline results will cause the model to avoid generating the same segmentations in it although they may be correct already, e.g., "taste" to "flavors", "so far" to "up until now", as well as others in red. As a result, the fluency and accuracy of the final results may be affected. 2) Equipped with DTG, fluency, coherence and grammatical correctness of generated results are all promoted. In the first case, the DTG result is more faithful not only in semantics but also in structure than the baseline. In the second case, DTG is able to complete the subject "I" which does not appear in the source sentence.

E Details of Error Statistical

In Figure 6, two types of error are considered (i.e., under translation and entity translation error). In this section, we provide the details of the method to conduct the error statistics.

Under Translation We first use *awesome-align*[11] to get the alignment between the source and target sentences. Then, a word in the source sentence is regarded as under translation, when it is aligned to a word in the reference target sentence but failed to be aligned in the generated target sentence.

Entity Translation We first use *spaCy*[12] to recognize the named entities in the reference target sentence, where person names, organizations and locations are considered. Then, an entity in the reference is considered an error if it cannot be found in the generated target sentence.

[11] https://github.com/neulab/awesome-align
[12] https://github.com/explosion/spaCy